Data-driven models for cell motility in complex 2- and 3-dimensional environments

Thesis submitted in accordance with the requirements of the University of Liverpool for the degree of Doctor in Philosophy

by Marianne Scott

October 2021

Acknowledgements

I would like to thank Prof Rachel Bearon and Dr Kamila Żychaluk for their continued support throughout my PhD and the completion of this work.

I am also grateful to all the collaborators that made this work possible through the provision of experimental data and useful discussions about how to build appropriate models and interpret the results.

For the work on glioblastoma cell motility I am thankful to Violaine See, Rosalie Richards, Dave Mason and Rebecca Kelly at the University of Liverpool for sharing their high quality data and all of their help with the project in its various stages. For the work on surface-attached bacteria I am grateful for the data and support provided by William Durham and Jamie Wheeler at the University of Sheffield.

Contents

Abstract

Data-driven models for cell motility in complex 2- and 3-dimensional environments

Marianne Scott

Studying cell motility is of vital importance for health, for knowing how cells behave and are affected by, and can themselves cause, disease. Mathematical modelling of such behaviour has proved beneficial for furthering knowledge of important motility processes in many different cell types. This work aims to define and analyse data-integrated mathematical models for cell motility in 2 and 3 dimensions, specifically applied to glioblastoma tumour cells and surface-attached *P. aeruginosa* bacterial cells. Models are outlined, tested on *in silico* data, parametrized where possible and assumptions are studied in detail. As a result, recommendations are made for how subsequent data could be collected to further improve the prediction and validation of these models.

A comprehensive framework is developed for the analysis of cell tracking data in 2 and 3 dimensions which allows a user to study various aspects of the Persistent Random Walk model as applied to these tracks, looking at speeds, persistence time, mean squared displacement and root mean squared speed. *In silico* simulations show good agreement with model predictions, however the model is incapable of describing the experimental data, as evidenced by lack of agreement in speed distributions and the speed parameter changing with time. A Bayesian approach to estimating these parameters is also considered, with estimates of persistence time seen to be inflated here compared to those from the frequentist approach.

A newly-observed twiddling mechanism used in chemotaxis by *P. aeruginosa* is also studied, through rigorous hypothesis testing of assumptions about this motion. An individual-based model is employed to simulate the resulting chemotactic motion, which shows good agreement with results from the specified analytic model, though the model cannot currently be validated against experimental data due to lack of appropriate data for parameter estimation.

Overview of the thesis

The importance of mathematical models, particularly those that make use of experimental data, has never been more recognized than in recent times. The power of the '*in silico*-first' approach along with the revolution in big data has meant that the use of mathematical and statistical tools are helping to further many research fields, one of those being the life sciences. Integrating data into models allows improvement of parameter estimates and gives the modeller the ability to validate their estimates and test hypotheses with experimentalists. This ultimately can save time and money in labs, but also can provide insights that may not have been revealed through practical experimentation alone.

There is real potential for an iterative working culture where experimentalists and modellers collaborate to develop sound mathematical models where assumptions are fully justified, models are appropriate for the data being studied and parameters are estimated along with some idea of the uncertainty surrounding them. There is real benefit to both parties working in this way, with experimentalists gaining insights that may not have been previously found and being able to explore ideas that may lie outside the possibilities of practical work, and modellers gaining new insights into their models, with realistic constraints on parameters and solutions.

One field in particular that has benefited from technological advances is that of cell motility. The ability to image cells at ever increasing resolutions and over more frequent time intervals requires mass data storage and with the advent of big data this has become possible. There is also increased capability for imaging live cells moving in 3 dimensions and being able to track their movement in all 3 of these spatial dimensions along with time. New sets of tracking data can thus

be collected from experiments and more detailed hypotheses can be tested from this data.

It is of vital importance that cell motility is studied, both from the perspective of disease and health. Finding out how cells move, be they eukaryotes or prokaryotes, healthy or pathogenic, can help us to understand how cells normally behave and thus how disease can affect them. From the pathogenic perspective we can also study how cells cause disease. All of this can point towards novel methods for disease and infection control and prevention, something which is highly desirable with the threat of antimicrobial resistance and in an age where cancer still kills 50% of those that get it in the UK (Cancer Research UK, 2021).

This thesis will look at modelling cell motility in both 2 and 3 dimensions and apply this in two cases; to cells of the brain cancer glioblastoma and bacterial cells from the strain *P. aeruginosa*. In both cases stochastic models are used in an attempt to model the motility of these cells, with data-integration being central to the approach. Along with parameter estimation and uncertainty quantification, suggestions are provided for the collection of future experimental data in order to allow improved fit of these models and consequently the parameter estimates as a result.

In chapters 1 and 2, a framework for modelling cell motility in 2 and 3 dimensions is outlined, with details about the stochastic model used - the Persistent Random Walk model - and the parametrization of this model along with confidence limits for estimated parameter values. Goodness-of-fit of this model is also tested with the use of various statistical measures, all of which take into account the correlation inherent in the data. The framework is tested with *in silico* data first before being applied to experimental data from glioblastoma tumour spheroids. This work has been pulished in Scott *et al.* (2021).

In chapter 3 this work is picked up again with parameter estimates being calculated using Bayesian methods instead of frequentist ones as in the previous chapters. In this work the focus is on estimating the parameters from the statistical measures mentioned above that test goodness-of-fit, providing a picture of uncertainty around these estimates and carrying out model selection to

investigate alternative models.

Finally, chapter 4 of this work focuses on modelling the motility of surface-attached *P. aeruginosa*, a pathogenic bacteria which carries out chemotaxis. This chapter will look at testing hypotheses about turning events seen during chemotaxis that are newly discovered and not yet understood. They are different in nature compared to those typically observed in, for example, *E. coli*, and so this work looks to answer questions about the potential bias caused by this motility mechanism and just how it differs from what has been previously seen. An individual-based model based on a velocity jump process is also defined and studied in detail, with the aim of parametrizing this model realistically and making use of experimental data for this purpose.

Chapter 1

A framework for modelling cell motility based on the Persistent Random Walk model: Part I - Introduction and Theory

1.1 Introduction and Literature Review

The ability of a cell to migrate is fundamental to its survival. Cells migrate through all manner of different environments, and in order to further our understanding of how systems in the body, both of humans and other organisms, function normally and under the influence of disease, we should endeavour to be able to describe cell motility under different conditions and rigorously test hypotheses about this motion. One way to do this is using mathematical models.

It is becoming increasingly evident that mathematical models can aid discovery in the life sciences, particularly when modelling complex phenomena such as cell migration and systems in which cells and their properties are being studied e.g. in cancer research (Deisboeck *et al.*, 2009; Anderson & Quaranta, 2008; Friedl *et al.*, 2012; Lee, 2018). To be predictive, mathematical models of cell migration should be informed by biology, dictating the relevant terms to be in-

cluded in a model, the initial and boundary conditions needed to constrain the system and providing model-specific values of important cell motility parameters. In return, these mathematical models can inform biology by analysing experimental data, confirming or rejecting proposed cell motility hypotheses, testing a system's sensitivity to model parameters, and being able to make quantitative predictions from numerous *in silico* simulations under different conditions. This can aid biologists in deciding which experiments may be useful for a study without wasting time, money or resources.

Much of the body of work concerning the study of cell motility includes studies which have been conducted in 2D, or on single cells in 3D. Due to the advent of advanced techniques in microscopy and *in vitro* models for studying cell motility in 3D, live 3D tracking of cells in tissues is now becoming increasingly possible (Hoarau-Véchot *et al.*, 2018; Yamada & Cukierman, 2007; Hakkinen *et al.*, 2011; Lee *et al.*, 2014; Paul *et al.*, 2016). This in turn has exposed major differences in the way that cells move in 3D environments compared to 2D (Wu *et al.*, 2018; Yamada & Cukierman, 2007; Antoni *et al.*, 2015), and how cells interact with their environment and each other, highlighting the need for new models of cell migration in 3D.

A major difference in 3D cell migration compared to 2D is the way that cells interact with each other and the extracellular matrix (ECM) surrounding them in many different ways. Further complexity arises because individual cells can behave very differently from each other in this environment. Because of the complexity of this 3D system and the potential for cellular heterogeneity, stochastic individual-based models capable of describing cells as individuals may be crucial to reveal the underlying mechanisms of cell motility in 3D.

The recently developed biological methods for studying cell motility produce large data sets in the form of cell tracks, and up to now there is a lack of mathematical tools to rigorously and systematically analyse this data, test proposed cell motility hypotheses, and compare this analysis across different models (Driscoll & Danuser, 2015; Friedl *et al.*, 2012). There are few mathematical models of 3D cell motility in existence, much less in number than their 2D counterparts,

though numerous biophysical models are found in the literature (Schlüter *et al.*, 2012; Paul *et al.*, 2017; Wu *et al.*, 2018). Rangarajan & Zaman (2008) provide a helpful review of existing mathematical models of 3D cell motility, which can be loosely categorised into force-based models, lattice-based models and stochastic models.

Force-based models focus on traction forces in cells due to the ECM and the protrusion of cells into it, as well as drag and adhesion forces that arise as a cell moves. Zaman *et al.* (2005, 2006) make use of such a model, calculating the forces on a cell at each time step in an attempt to describe the cell's motility as a function of time. Lattice-based Monte Carlo methods are based on a 3D lattice and a set of criteria which dictates a cell's movement at each time step (Zaman *et al.*, 2007).

Stochastic models are generally based around stochastic differential equations and random walks (Parkhurst & Saltzman, 1992; Wu *et al.*, 2015), the Persistent Random Walk (PRW) model being of particular interest to our current work. Wu *et al.* (2014) investigated the fit of the PRW model to 3D motility data, concluding that the model was incapable of describing motility in 3D, and adding an adjustment to the model in 2D to explain heterogeneity seen in experimental data. In a later work they propose the Anisotropic Persistent Random Walk (APRW) model which they claim better describes motility data in 3D with consideration of anisotropy in motility that the standard PRW model does not take into account (Wu *et al.*, 2015).

The PRW model has long been used to describe cell motility in 2D (Gail & Boone, 1970; Dunn & Brown, 1987; Stokes & Lauffenburger, 1991; Tranquillo & Lauffenburger, 1987; Dimilla *et al.*, 1992), though many have questioned whether the statistical measures defined by the model actually fit experimentally collected data. Most commonly these studies find that the Mean Squared Displacement (MSD) of cells is found to follow a power law rather than being a linear function of time as the PRW predicts (Dieterich *et al.*, 2008; Upadhyaya *et al.*, 2001; Metzner *et al.*, 2015; Loosley *et al.*, 2015; Cherstvy *et al.*, 2018). The Velocity Autocorrelation Function ($VACF$) is found to be better modelled by a sum of

two exponentials rather than a single exponential (Dieterich *et al.*, 2008; Wu *et al.*, 2014) and non-Gaussian distributions in cell velocities are found in some studies (Dieterich *et al.*, 2008; Metzner *et al.*, 2015). These model properties are discussed in more detail below. Some studies have shown that cells migrating in 3D, particularly cancer cells, display sub- or superdiffusive behaviour (Yurchenko *et al.*, 2019; Luzhansky *et al.*, 2018; Takagi *et al.*, 2008), meaning the PRW model description of the MSD would over- or underestimate this quantity for a population of cells.

Nevertheless, the PRW model is historically one of the most widely used models of cell motility and we use it here to demonstrate the power and usability of our framework. We provide mathematical tools to analyse 2D and 3D cell tracking data, using statistical measures to validate the model and provide parameter estimates to allow for its parametrisation in specific cases.

We believe the framework is adaptable and the description presented in this thesis is meant as a starting point to demonstrate a rigorous protocol for analysis. Whilst our framework is based on the PRW model, we present it as a method for analysing cell tracks, easily adapted to different models and the inclusion of biologically-informed terms in the governing equation of a model.

We first carry out *in silico* simulations of the model to build the framework and then test it using experimental data from glioblastoma (GBM) cell tracks *in vitro*, which for the 3D case is taken from a subset of the data found in Richards *et al.* (2018), and for the 2D case was taken from experimental data collected by students from the same team.

GBM is a particularly fatal brain tumour for which treatment methods inevitably fail due to the highly proliferative and invasive nature of the cells. The recent rise of the field of mathematical oncology (Rockne *et al.*, 2019) has seen mathematical models attempt to describe many different aspects of cancer. This area of research aims to use mathematical models to assist in the fight against cancer, a disease which is characterised by excessive cell motility, especially invasion of cells into healthy tissue. Improving our understanding of cell motility will thus likely improve mathematical models in this field and eventually lead

to better outcomes for patients with fatal brain tumours like GBM. Models of tumours in 3D are becoming increasingly predictive due to data-integration and increased knowledge of the tumour microenvironment that comes with an ability to replicate experimentally the conditions found in this environment.

Many models of tumours in 3D are found in the literature, together describing a range of features of tumours in a 3D environment. Data-integrated continuum models are a popular choice due to the wide range of analytical tools available for investigating these systems. The ability to integrate experimental data into these models makes them suitable for predicting survival times and potential treatment regimens for individuals (Hathout *et al.*, 2016; Swanson *et al.*, 2008; Colombo *et al.*, 2015; Rockne *et al.*, 2015; Agosti *et al.*, 2018; Jackson *et al.*, 2015). However, these continuous models are incapable of modelling individual cells in a tumour, and due to the inherently stochastic nature of cell motility, and cancer in particular, it is evident that discrete, stochastic models will be needed to further this field of study.

Stochastic models of tumours and cancer cells broadly fall into one of two categories: agent-based models which can be on- (Gerlee & Nelander, 2012; Hamis *et al.*, 2019; Scianna & Preziosi, 2014) or off-lattice (Lowengrub *et al.*, 2010; Macklin *et al.*, 2010) and those based on stochastic differential equations and random walks (Stein *et al.*, 2007; Antonopoulos & Stamatakos, 2015; Antonopoulos *et al.*, 2019; Wu *et al.*, 2015), both of which attempt to use the properties of individual cells to elucidate the population behaviour under different conditions. We note that cell-based and continuum models of cell motility can be connected using scaling techniques, as described in Othmer & Xue (2013), for example.

The first two chapters in this thesis will cover work completed in 2 and 3 dimensions using the Persistent Random Walk model and its application to cell tracking data from glioblastoma tumour cells. The first of the chapters will provide an overview of the PRW model in all 3 physical dimensions, including model background and detailed consideration of statistical measures used both to test the goodness-of-fit and estimate model parameters in the relevant PRW model. Chapter 2 will set out the framework created in MATLAB to test cell

motility hypotheses in 2 and 3 dimensions and will provide examples of the testing of this framework using *in silico* data sets and its subsequent application to 2- and 3-dimensional experimental data sets.

1.2 The Persistent Random Walk Model - Background

The Persistent Random Walk (PRW) model has long been used as a way to describe random cell motility. The model, derived from the stationary, mean-reverting Ornstein-Uhlenbeck (OU) process (Dunn & Brown, 1987), describes a correlated random walk in velocity which sees the correlation between subsequent velocities of the same cell decay over time. A cell's velocity in a subsequent time step is assumed to be conditional on the velocity in the current time step, with past velocities having no influence, and tends to be in the same direction. Cells are assumed to be identical, and independent - no interaction between cells is modelled.

1.2.1 The Fokker-Planck Equation and the Stochastic Differential Equation in relation to diffusion processes

A diffusion process such as cell motility takes place over time and the evolution of this process over time is described by a stochastic differential equation (SDE). An SDE for a process with n variables has the general form

$$d\mathbf{x} = \mathbf{A}(\mathbf{x}, t)\, dt + \boldsymbol{B}(\mathbf{x}, t)\, d\mathbf{W}(t), \tag{1.1}$$

where $\mathbf{x}$ is a vector of n random variables of interest, t is time, $\mathbf{A}(\mathbf{x}, t)$ is the drift vector which encompasses the deterministic part of the SDE, $\boldsymbol{B}(\mathbf{x}, t)$ is the diffusion matrix which incorporates the random noise and $\mathbf{W}(t)$ is an n-dimensional Wiener process. The solution of the SDE gives a particular stochastic process $\mathbf{x}(t)$ which we can evaluate at any time t for a given realisation of noise.

We can get a family of solutions to the SDE by looking at the closely related Fokker-Planck (F-P) equation; an equation for a conditional probability density $p = p(\mathbf{x}, t|\mathbf{x}_0, t_0)$ which describes the probability of the cell being at position $\mathbf{x}$ at time t given that they were initially at $\mathbf{x}_0$ at time t_0 (Gardiner, 2009). The general form of the F-P equation for a stochastic process with n variables is

$$\frac{\partial p}{\partial t} = -\sum_i \frac{\partial}{\partial x_i}\left(A_i(\mathbf{x}, t)p\right) + \frac{1}{2}\sum_{i,j}\frac{\partial}{\partial x_i \partial x_j}\left(\left[\boldsymbol{B}(\mathbf{x}, t)\boldsymbol{B}^T(\mathbf{x}, t)\right]_{ij} p\right), \tag{1.2}$$

where x_k is the k^{th} element of vector $\mathbf{x}$, $A_i(\mathbf{x}, t)$ is the i^{th} component of the drift vector $\mathbf{A}$ and $\boldsymbol{B}(\mathbf{x}, t)$ is the diffusion matrix. The F-P equation looks at the time evolution of the probability density function for $\mathbf{x}$ and thus its solution gives the density function for $\mathbf{x}$ at time t.

1.2.2 The Wiener Process

The Wiener process which appears in equation 1.1, is a stochastic, Markov diffusion process used to represent noise in a stochastic model. Any diffusion process can be expressed in terms of the Wiener process through a SDE. Studied first by Norbert Wiener, the density of the process is obtained from the solution to the F-P equation with only one variable $W(t)$ which has drift coefficient 0 and diffusion coefficient 1 (Gardiner, 2009), written as

$$\frac{\partial}{\partial t}p(w, t|w_0, t_0) = \frac{1}{2}\frac{\partial^2}{\partial w^2}p(w, t|w_0, t_0), \tag{1.3}$$

where $p(w, t|w_0, t_0)$ is the conditional probability distribution of $W(t) = w$ (which is the variable of interest) at time t, given $W(t_0) = w_0$.

In its one dimensional form, the Wiener process it is often referred to as Brownian motion since this type of motion also satisfies equation 1.3. The resulting solution of the F-P equation shows that the Wiener process has a Gaussian distribution with mean w_0 and variance $t - t_0$.

A multivariate, or n-dimensional, Wiener process

$$\mathbf{W}(t) = \left[W_1(t), W_2(t), \ldots, W_n(t)\right],$$

where $W_i(t)$ is a 1-dimensional Wiener process, and its density is obtained from

solving the multivariate F-P equation

$$\frac{\partial}{\partial t}p(\mathbf{w}, t|\mathbf{w_0}, t_0) = \frac{1}{2}\sum_i \frac{\partial^2}{\partial w_i^2}p(\mathbf{w}, t|\mathbf{w_0}, t_0),$$

again with diffusion coefficients equal to 1 and drift coefficients equal to 0, leading to the process having a multivariate Gaussian distribution where

$$\langle \mathbf{W}(t) \rangle = \mathbf{w}_0$$

and

$$\langle [W_i(t) - w_{0i}][W_j(t) - w_{0j}] \rangle = (t - t_0)\delta_{ij},$$

with $W_i(t)$ being the i^{th} element of $\mathbf{W}(t)$, $w_{0i} = W(t_{0i})$, $\langle . \rangle$ indicates taking the expectation and δ_{ij} is the Kronecker delta function taking values

$$\delta_{ij} = \begin{cases} 0, & i \neq j \\ 1, & i = j. \end{cases}$$

It is also important to see that the Wiener process has independent increments, such that for times s_1, s_2, t_1 and t_2, if $0 \leq s_1 < t_1 \leq s_2 < t_2$ then $W(t_1) - W(s_1)$ and $W(t_2) - W(s_2)$ are independent random variables. This extends to n independent random variables resulting from n increments, say, between t_1 and s_1 and t_n and s_n.

1.2.3 The Ornstein-Uhlenbeck Process and the Persistent Random Walk model

The Persistent Random Walk (PRW) model (Dunn & Brown, 1987) describes diffusion in the velocity space and is derived from the Ornstein-Uhlenbeck (OU) process (Uhlenbeck & Ornstein, 1930). The OU process is a stationary Gauss-Markovian stochastic process which is more apt for describing particle velocities than the Wiener process, due to the Wiener process allowing particle speeds to be infinite and this not being physically realistic.

The OU process is also a diffusion process, and its distribution is obtained upon solving the F-P equation 1.3 used for the Wiener process with the addition of a linear drift term with constant coefficient. The OU process is thus a

mean-reverting process as the drift term is included. The process will eventually converge to its stationary mean with a non-zero variance in the limit as $t \to \infty$.

In 1D, the probability density function $p(v,t)$ of velocity v at time t, governed by the OU process, can be described by the Fokker-Planck equation

$$\frac{\partial p}{\partial t} = \frac{\partial(\beta v p)}{\partial v} + \frac{1}{2}\alpha \frac{\partial^2 p}{\partial v^2}, \tag{1.4}$$

where v is the cell's velocity at time t, $p = p(v,t|v_0,0)$ is the probability density function of v given $v = v_0$ at $t = 0$, α is the diffusion coefficient which represents the magnitude of random movement accelerations, and β is the drift coefficient which represents the velocity decay rate.

The time evolution of this OU process can be described by the stochastic differential equation (SDE) for cell velocity v

$$dv = -\beta v\, dt + \sqrt{\alpha}\, dW(t),$$

where $W(t)$ is the Wiener process. In 1D, for an initial distribution of velocity taking the value v_0 with probability one, the solution of this equation is a Gaussian distribution with mean $\mu = v_0 e^{-\beta t}$ and variance $\sigma^2 = \frac{\alpha}{2\beta}(1 - e^{-2\beta t})$.

Statistical analysis of the properties of the OU process reveals that two experimentally relevant quantities, the root mean squared speed ($RMSS$) of cells at steady state, S and persistence time, P, can be used in the SDE to model velocity, giving rise to the 2D PRW model as in Stokes & Lauffenburger (1991). They express α and β in terms of these more intuitive parameters giving $P = 1/\beta$ and $S = \sqrt{\alpha/\beta}$.

More generally, in n dimensions, Campos *et al.* (2010) express the process in terms of the persistence time P, and D_n, the spatial diffusion coefficient of the cells in n-dimensional physical space, giving the n-dimensional SDE for the PRW model as

$$d\mathbf{v} = -\frac{1}{P}\mathbf{v}\, dt + \frac{\sqrt{2D_n}}{P}\, d\mathbf{W}(t), \tag{1.5}$$

where $\mathbf{v}$ is the n-dimensional velocity vector, $\mathbf{W}(t)$ is an n-dimensional Wiener process (an n-dimensional vector of 1-dimensional Wiener processes) and $D_n = S^2 P/n$ is the n-dimensional diffusion coefficient.

We can see from equation 1.5 that acceleration is linear and only depends on velocity and some random noise from the Wiener process.

It is also possible to derive this SDE from the equation of motion $F = ma$. If we take force to be $F = -kv$ where F is friction given by the product of some positive constant k and velocity v, and $a = \frac{dv}{dt}$, then we have

$$\begin{aligned} \frac{dv}{dt} &= -\frac{kv}{m} \\ \implies dv &= -\frac{v}{P}\,dt \\ \implies dv &= -\frac{v}{P}\,dt + \text{noise} \end{aligned}$$

if in the last step we simply add white noise to convert this equation into a stochastic one.

Considering what happens in the limit, as $P \to 0$ this simply becomes Brownian motion, completely random motion, with infinite negative acceleration and lots of noise. As $P \to \infty$ the opposite situation arises, with acceleration being 0, i.e., velocity is constant, and noise will disappear, thus giving perfect rectilinear motion.

1.3 Statistical Measures

To decide on whether the PRW model is an appropriate model for a given data set and, if this is the case, to estimate the values of S and P, we implement statistical measures. Such statistical measures are drawn from the model using equation 1.5.

The first such measure, the Velocity Autocorrelation Function ($VACF$), is given for the PRW model in n dimensions at time t by (Campos *et al.*, 2010)

$$VACF(t) = \frac{nD_n}{P} e^{-\frac{t}{P}} = S^2 e^{-\frac{t}{P}}. \tag{1.6}$$

The $VACF$ quantifies the correlation between cell velocity at time 0 and at time t. This is calculated at a population level, averaging over all cells for each time. The correlation decays at rate $1/P$, meaning that cells 'forget' their previous velocity over times long compared with P.

Secondly, the Mean Squared Displacement (MSD) which is commonly used when looking at cell motility, is given for the PRW model in n dimensions by (Campos *et al.*, 2010)

$$MSD(t) = 2nD_nP(e^{-\frac{t}{P}} + \frac{t}{P} - 1) = 2S^2P^2(e^{-\frac{t}{P}} + \frac{t}{P} - 1). \tag{1.7}$$

We see the PRW model displays classic diffusion behaviour of MSD tending to a linear function of time, i.e. $MSD(t) \to 2S^2Pt$ as $t \to \infty$.

Finally, the stationary speed distribution of the population is considered. At steady state, velocities should follow an n-dimensional Gaussian distribution according to the PRW model. For 3D this implies that the speed, u, follows a Maxwell-Boltzmann distribution with density

$$f(u; S) = \left(\frac{3}{2\pi S^2}\right)^{3/2} 4\pi u^2 e^{-\frac{3u^2}{2S^2}}, \tag{1.8}$$

and in 2D speeds should follow the Rayleigh distribution with density

$$f(u; S) = \frac{2u}{S^2} e^{-\frac{u^2}{S^2}}.$$

More detailed derivations of these quantities are provided now, putting each measure in context and providing full reasoning behind the use of each one.

1.3.1 Velocity Autocorrelation Function

Time correlation function for the OU process

We can define the time correlation for the OU process in velocity, as in (Gardiner, 2009), as

$$\begin{aligned}\langle V(t)\cdot V(s)|[v_0, t_0]\rangle &= \int\int v_1\, v_2\, p(v_1, t; v_2, s|v_0, t_0)\, dv_1\, dv_2 \\ &= \int\int v_1\, v_2\, p(v_1, t|v_2, s)\, p(v_2, s|v_0, t_0)\, dv_1\, dv_2,\end{aligned} \tag{1.9}$$

where $V(t)$ and $V(s)$ are the random variables describing the velocities at times t and s and we are assuming $t \geq s \geq t_0$. $p(v_1, t; v_2, s|v_0, t_0)$ is the conditional probability density function of a cell having velocity v_1 at time t and velocity v_2 at time s given that it had velocity v_0 at time t_0.

Using the Markov property of the OU process, we see that $p(v_1, t|v_2, s; v_0, t_0) = p(v_1, t|v_2, s)$; that is, the conditional probability density function of a cell having velocity v_1 at time t given it had velocity v_0 at time t_0 and velocity v_2 at time s, only depends on the value of velocity at time s (Gardiner, 2009). We then can write

$$p(v_1, t; v_2, s|v_0, t_0) = p(v_1, t|v_2, s; v_0, t_0)p(v_2, s|v_0, t) = p(v_1, t|v_2, s)p(v_2, s|v_0, t).$$

Taking equation 1.4, but using $\alpha = k$ and $\beta = D$ as in Gardiner (2009), we will derive an expression for the time correlation of the OU process in terms of cell velocity v, by solving the F-P equation

$$\frac{\partial p}{\partial t} = \frac{\partial (kvp)}{\partial v} + \frac{1}{2}D\frac{\partial^2 p}{\partial v^2}, \tag{1.10}$$

for $p(v, t)$ to obtain the probability density functions that the formula in equation 1.9 requires.

We look for a stationary solution where $\frac{\partial p(v,t)}{\partial t} = 0$ for velocity v and time t, and $p(v, t)$ satisfies equation 1.10. The subscript s in the following derivations thus represents the stationary nature of these calculations. This stationary property along with equation 1.10 means we have

$$\frac{\partial}{\partial v}\left(kvp + \frac{1}{2}D\frac{\partial p}{\partial v}\right) = 0. \tag{1.11}$$

We impose the boundary condition $p(-\infty, t) = 0$ and since $p(v, t)$ is smooth, we can also impose that $\frac{\partial}{\partial v}p(-\infty, t) = 0$. Integrating equation 1.11 with respect to v from $-\infty$ to v gives

$$\left[kvp + \frac{1}{2}D\frac{\partial p}{\partial v}\right]_{-\infty}^{v} = 0,$$
$$\implies kvp + \frac{1}{2}D\frac{\partial p}{\partial v} = 0$$

which after some rearranging becomes

$$\frac{1}{p}\frac{\partial p}{\partial v} = -\frac{2kv}{D}.$$

Solving this as a first order differential equation for $p(v)$ by separation of variables yields

$$p(v) = Ae^{-\frac{kv^2}{D}}, \tag{1.12}$$

where A is a multiplicative constant. As p is a probability density function, A needs to be such that $\int_{-\infty}^{\infty} p(v)dv = 1$. From equation 1.12, we can see that $p(v)$ is a Gaussian distribution with mean 0 and variance $D/2k$, and thus $A = \left(\frac{\pi D}{k}\right)^{-\frac{1}{2}}$ (Gardiner, 2009).

We can now look at how $p(v)$ should be used in the time correlation function given in equation 1.9 to derive the formula we require. We also note here that we are interested in the stationary correlation function which will be independent of time.

With a view to obtaining the stationary correlation function, we first impose the initial condition on $p(v_2, s|v_0, t_0)$ far back into the past by letting $t_0 \to -\infty$ so it is far from s and thus the cell can't remember its velocity v_0 at t_0. We again get independence of v_0 and this gives

$$\lim_{t_0 \to -\infty} p(v_2, s|v_0, t_0) = p_{\mathrm{S}}(v_2) = \left(\frac{\pi D}{k}\right)^{-\frac{1}{2}} e^{-\frac{k{v_2}^2}{D}}, \tag{1.13}$$

and so $p_{\mathrm{S}}(v_2)$ is normally distributed with mean 0 and variance $D/2k$.

To obtain the stationary probability distribution for $p(v_1, t|v_2, s)$ we must delve a little deeper into the F-P equation for the OU process. In this case, we want to find a solution to the F-P equation which has a distribution of the form $p(v, t|v_0, 0)$, one which is still dependent on t but possesses the Markovian property. For this we need to use a characteristic function, defined as (Gardiner, 2009)

$$\phi(s) = \left\langle e^{i\mathbf{s}\cdot\mathbf{X}} \right\rangle = \int p(\mathbf{x}) e^{i\mathbf{s}\cdot\mathbf{x}}\, d\mathbf{x},$$

for some random variable X and in our specific case defined as

$$\phi(v, t) = \int_{-\infty}^{\infty} e^{isv} p(v, t|v_0, 0)\, dv,$$

to transform the F-P equation, equation 1.10.

We then must solve the corresponding transformed F-P equation

$$\frac{\partial \phi}{\partial t} + ks\frac{\partial \phi}{\partial s} = -\frac{1}{2} D s^2 \phi,$$

using the method of characteristics to obtain the solution

$$\phi(v, t) = \exp\left[-\frac{Ds^2}{4k}\left(1 - e^{-2kt}\right) + isv_0 e^{-kt}\right].$$

It then remains to transform back into p using the transform

$$p(\mathbf{v}) = \frac{1}{(2\pi)^n} \int \phi(\mathbf{s}) e^{-i\mathbf{v}\cdot\mathbf{s}}\, d\mathbf{s},$$

and we get

$$p(v, t|v_0, 0) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(v-\mu)^2}{2\sigma^2}}, \tag{1.14}$$

which is Gaussian with mean $\mu = v_0 e^{-kt}$ and variance $\sigma^2 = \frac{D}{2k}\left(1 - e^{-2kt}\right)$.

Noting that we are looking for the stationary distribution, we can write $p(v_1, t|v_2, s) = p(v_1, t-s|v_2, 0)$ and so putting $v = v_1$, $t = \tau$ and $v_0 = v_2$ into equation 1.14, we get the stationary distribution $p_{\mathrm{S}}(v_1)$ with the same form but with mean $\mu = v_2 e^{-k\tau}$ and variance $\sigma^2 = \frac{D}{2k}\left(1 - e^{-2k\tau}\right)$, where τ is known as the correlation time (Gardiner, 2009) or time lag, and is defined as

$$\tau = \int_0^\infty \langle V(t), V(0)\rangle_{\mathrm{S}} / \mathrm{Var}[V]_{\mathrm{S}}\, dt \equiv 1/k \sim |t-s|. \tag{1.15}$$

We can thus substitute both of the stationary distributions from equations 1.13 and 1.14 into the time correlation function expression 1.9 to obtain

$$\begin{aligned}
\langle V(t)\cdot V(s)|[v_0, t_0]\rangle_s &= \int\int v_1\, v_2\, p(v_1, t|v_2, s)\, p(v_2, s|v_0, t_0)\, dv_1\, dv_2 \\
&= \int v_2\, p(v_2, s|v_0, t_0) \int v_1\, p(v_1, t|v_2, s)\, dv_1\, dv_2 \\
&= \int \mu\, v_2\, p_{\mathrm{S}}(v_2)\, dv_2 \\
&= \int v_2\, e^{-k\tau} \cdot v_2\, p_{\mathrm{S}}(v_2)\, dv_2 \\
&= e^{-k\tau} \int v_2^2\, p_{\mathrm{S}}(v_2)\, dv_2 \\
&= e^{-k\tau} \cdot \mathrm{Var}[p_{\mathrm{S}}(v_2)] \\
&= e^{-k\tau} \cdot \frac{D}{2k}
\end{aligned}$$

Thus the time correlation function in velocity is finally given by

$$\langle V(t)\cdot V(s)\rangle_{\mathrm{S}} = \frac{D}{2k} e^{-k\tau}. \tag{1.16}$$

Velocity Autocorrelation function for the PRW model

Our 1-dimensional $VACF$, eliminating the subscript s from now on, is

$$\langle v(t)\cdot v(s)\rangle = \frac{D}{2k} e^{-k\tau}. \tag{1.17}$$

We can then define our persistence time parameter P intuitively from this expression, noting that it is the timescale of the exponential decay and is thus equal to $1/k$.

We can also gain a measure of the cell speed from this expression. If $t = s$, we can see from 1.15 and 1.17 that we would obtain the mean squared speed of a cell, giving

$$\langle v(t) \cdot v(t) \rangle = \left\langle v(t)^2 \right\rangle = \frac{D}{2k},$$

and so we define $D/2k$ as the mean squared speed S^2, obviously taking the square root to obtain parameter S, defined as the root mean squared speed.

Studying this velocity autocorrelation function closer, we see that

$$\lim_{\tau \to \infty} e^{-\frac{\tau}{P}} = 0,$$

confirming that over longer time periods the correlation between the velocities gets worse as they 'forget' where they were and how fast they were going. Looking also at the limit of persistence

$$\lim_{P \to \infty} e^{-\frac{\tau}{P}} = 1,$$

we see that the longer a cell persists in a certain direction, the better the correlation between the velocities, as expected and in complete contrast to what would be a zero correlation if the persistence time tended to 0 and the cell was changing direction very frequently.

Higher Dimensional $VACF$

We now consider the $VACF$ in 2 and 3 dimensions. We note that the 2 dimensional governing equation is

$$d\mathbf{v} = -\frac{1}{P}\mathbf{v}\, dt + \frac{S}{\sqrt{P}}\, d\mathbf{W}(t), \tag{1.18}$$

where $\mathbf{v} = (v_x, v_y)^T$ using the components of velocity in the x- and y- directions and $\mathbf{W(t)} = (W_1(t), W_2(t))^T$ is a bivariate Wiener process with 1-dimensional Wiener processes $W_1(t)$ and $W_2(t)$. A similar equation can be written in 3 dimensions if we include the z-direction in equation 1.18, where $\mathbf{v} = (v_x, v_y, v_z)^T$

and $\mathbf{W}(t) = (W_1(t), W_2(t), W_3(t))^T$ is a 3-dimensional Wiener process with 1-dimensional Wiener processes $W_1(t)$, $W_2(t)$ and $W_3(t)$.

Realising that the $VACF$ itself is the expected value of the scalar product of the velocities being studied, to obtain the autocorrelation for time lag τ, we can write this product in 1 dimension as

$$VACF(\tau) = \langle v(t) \cdot v(t+\tau) \rangle .$$

Extending this to two dimensions, we have the expression

$$VACF(\tau) = \langle v_x(t) \cdot v_x(t+\tau) + v_y(t) \cdot v_y(t+\tau) \rangle ,$$

and for three dimensions

$$VACF(\tau) = \langle v_x(t) \cdot v_x(t+\tau) + v_y(t) \cdot v_y(t+\tau) + v_z(t) \cdot v_z(t+\tau) \rangle ,$$

for v_x, v_y and v_z being the x-, y- and z-components of the velocity.

From our above derivation, we know that in 1 dimension

$$VACF(\tau) = \langle v(t) \cdot v(t+\tau) \rangle = S^2 e^{-\frac{\tau}{P}},$$

so we can get expressions for the $VACF$ in 2 and 3 dimensions similarly as we expect the correlation of velocity to be of the same form in each of the x-, y- and z-directions. Given this, the subsequent algebra is omitted, but one can find an expression for the n-dimensional velocity autocorrelation function in Campos *et al.* (2010) as

$$VACF(\tau) = \frac{nD_n}{P} e^{-\tau/P}, \tag{1.19}$$

where S and P are as defined above and $D_n = S^2 P/n$. This means upon substitution of relevant n and D, we find that both the 2- and 3-dimensional velocity autocorrelation functions are in fact the same as that in 1 dimension.

1.3.2 Mean Squared Displacement

The most common measure of random cell movement is the mean squared displacement (MSD), or the average displacement of the cell from its initial position, evaluated at different time lags. This requires us to look at the displacement

between each time point in the track and calculate the expected value of these displacements squared. Let $x(t)$ be the x-position of a cell at time t. Then over the time interval $[0,\, t]$, the displacement of the cell is given by $x(t) - x(0)$. For a 3-dimensional system, the MSD of a cell at time t is given by

$$\begin{aligned} MSD(t) &= \left\langle (x(t) - x(0))^2 + (y(t) - y(0))^2 + (z(t) - z(0))^2 \right\rangle \\ &= \left\langle (x(t) - x(0))^2 \right\rangle + \left\langle (y(t) - y(0))^2 \right\rangle + \left\langle (z(t) - z(0))^2 \right\rangle, \end{aligned} \tag{1.20}$$

where $\langle . \rangle$ denotes taking the expected value and $x(t) - x(0)$, $y(t) - y(0)$ and $z(t) - z(0)$ are the total displacements of the cell in the x-, y- and z-directions respectively.

First considering only 1 dimension, we can derive an expression for the MSD which we would expect to get from the PRW model. Remembering that the displacement of the cell at time t in the x-direction is given by $x(t) - x(0)$, we can write this displacement over the interval $[0,\, t]$ in terms of the cell's velocity as

$$x(t) - x(0) = \int_0^t v(s)\, ds, \tag{1.21}$$

where $v(s)$ is the velocity of the cell at time s.

The MSD in 1 dimension is given by

$$MSD(t) = \left\langle (x(t) - x(0))^2 \right\rangle,$$

from equation 1.20 and restricting to only 1 dimension. To be able to calculate this quantity, we need to know the distribution of $x(t) - x(0)$, i.e. to determine the mean and variance of $x(t) - x(0)$. To do so, we first remember that velocity follows the OU process and so $\langle v(t) \rangle = 0$. Intuitively this is correct since the cells are equally likely to move either left or right without drift, if there is no initial bias.

We also define $\sigma_0^2 = \left\langle v(t)^2 \right\rangle$ as the variance of the velocity $v(t)$. Then, using our assumptions and equation 1.21 we have

$$\langle x(t) - x(0) \rangle = \int_0^t \langle v(s) \rangle\, ds = 0,$$

and, for $t > 0$

$$\begin{aligned}\left\langle (x(t) - x(0))^2 \right\rangle &= \int_0^t \int_0^t \langle v(s)v(s') \rangle \, ds \, ds' \\ &= \sigma_0^2 \int_0^t \int_0^t e^{-\beta|s-s'|} \, ds \, ds' \end{aligned} \tag{1.22}$$

using the linearity of expectation.

To calculate this integral we must notice that

$$|s - s'| = \begin{cases} s - s', & s > s' \\ s' - s, & s < s' \end{cases}$$

and so equation 1.22 becomes

$$\begin{aligned}\sigma_0^2 \int_0^t \int_0^t e^{-\beta|s-s'|} ds \, ds' &= \sigma_0^2 \int_{s'=0}^t \left(\int_{s=0}^{s'} e^{-\beta(s'-s)} ds + \int_{s=s'}^t e^{-\beta(s-s')} ds \right) ds' \\ &= \sigma_0^2 \int_{s'=0}^t \left(\left[\frac{1}{\beta} e^{-\beta(s'-s)} \right]_0^{s'} + \left[-\frac{1}{\beta} e^{-\beta(s-s')} \right]_{s'}^t \right) ds' \\ &= \frac{\sigma_0^2}{\beta} \int_{s'=0}^t \left(2 - e^{-\beta s'} - e^{-\beta(t-s')} \right) ds' \\ &= \frac{\sigma_0^2}{\beta} \left[2s' + \frac{1}{\beta} e^{-\beta s'} - \frac{1}{\beta} e^{-\beta(t-s')} \right]_0^t \\ &= \frac{\sigma_0^2}{\beta} \left(2t + \frac{2}{\beta} e^{-\beta t} - \frac{2}{\beta} \right) \\ &= \frac{2\sigma_0^2}{\beta^2} \left(\beta t + e^{-\beta t} - 1 \right). \end{aligned}$$

We know from looking at the $VACF$ that the variance of the velocity distribution is $\sigma_0^2 = S^2$. Thus using $\beta = 1/P$, in 1 dimension, we can write the MSD as

$$MSD(t) = 2S^2 P^2 \left(e^{-t/P} + \frac{t}{P} - 1 \right). \tag{1.23}$$

Clearly this expression is not dependent on dimension and remains the same in all 3 dimensions considered.

1.3.3 Speed and Velocity Distributions

Gaussian distribution of velocities

Another feature of the PRW model is a steady state Gaussian distribution of velocities (Wu *et al.*, 2014; Selmeczi & Mosler, 2005). We have seen from the

derivation of the $VACF$ that the probability density of position, velocity, or indeed any quantity satisfying the F-P equation 1.11, is Gaussian.

To see how this property comes about in 2 and 3 dimensions, we must solve the corresponding F-P equations. To obtain the 2-dimensional F-P equation we require, we use equation 1.2 with variables x and y to extend the 1D F-P equation 1.10. We still have diffusion coefficient D, though this time diffusion is in both the x and y directions, and the drift vector is $\mathbf{A} = [-kx, -ky]$. We thus obtain

$$\frac{\partial p}{\partial t} = \frac{\partial (kxp)}{\partial x} + \frac{\partial (kyp)}{\partial y} + \frac{1}{2} D \left(\frac{\partial^2 p}{\partial x^2} + \frac{\partial^2 p}{\partial y^2} \right). \tag{1.24}$$

We again look for a steady state solution when aiming to obtain the steady state Gaussian distribution p, so letting $\frac{\partial p}{\partial t} = 0$, equation 1.24 becomes

$$\begin{aligned} \frac{\partial (kxp)}{\partial x} + \frac{\partial (kyp)}{\partial y} + \frac{1}{2} D \left(\frac{\partial^2 p}{\partial x^2} + \frac{\partial^2 p}{\partial y^2} \right) &= 0 \\ \implies \left(\frac{\partial (kxp)}{\partial x} + \frac{1}{2} D \frac{\partial^2 p}{\partial x^2} \right) + \left(\frac{\partial (kyp)}{\partial y} + \frac{1}{2} D \frac{\partial^2 p}{\partial y^2} \right) &= 0. \end{aligned} \tag{1.25}$$

It is clear now that we have two 1-dimensional F-P equations which as we know are solved by the Gaussian probability density function. This prompts us to propose a solution that looks like a general bivariate Gaussian distribution. The general form of a n variable Gaussian probability density function is given by

$$p(\mathbf{x}) = \frac{1}{\sqrt{(2\pi)^n \det(\sigma)}} e^{-\frac{1}{2}(\mathbf{x}-\bar{\mathbf{x}})^T \sigma^{-1} (\mathbf{x}-\bar{\mathbf{x}})}, \tag{1.26}$$

where $\mathbf{x}$ is the vector of n variables, $\det(\sigma)$ is the determinant of the covariance matrix of $\mathbf{x}$ and $\bar{\mathbf{x}}$ is the vector of mean values for each of the n variables (Gardiner, 2009).

In our bivariate case, we will use variables x and y, both having mean value 0 and variance $D/2k$ as above, and note that they are independent of one another due to them being the axes on which the cells are moving. This means their correlation is 0 and σ would be a 2×2 matrix with only its diagonal elements being non-zero and each taking the value of the variance $D/2k$, i.e.

$\sigma = \begin{bmatrix} D/2k & 0 \\ 0 & D/2k \end{bmatrix}$. Letting $\mathbf{x} = \begin{bmatrix} x \\ y \end{bmatrix}$, and $\det(\sigma) = \left(\frac{D}{2k}\right)^2$, we can use equation 1.26 to obtain

$$\begin{aligned} p(x,y) &= \frac{1}{\sqrt{(2\pi)^2\left(\frac{D}{2k}^2\right)}} e^{-\frac{1}{2}\begin{bmatrix} x & y \end{bmatrix}\begin{bmatrix} 2k/D & 0 \\ 0 & 2k/D \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix}} \\ &= \frac{1}{\sqrt{(2\pi)^2\left(\frac{D}{2k}\right)}} e^{\begin{bmatrix} x & y \end{bmatrix}\begin{bmatrix} -k/D & 0 \\ 0 & -k/D \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix}} \\ &= \frac{k}{\pi D} e^{-\frac{k}{D}(x^2+y^2)}. \end{aligned} \tag{1.27}$$

Since x and y are independent, we can split their joint distribution into the distribution of x and the distribution of y as follows;

$$\begin{aligned} p(x,y) &= \frac{k}{\pi D} e^{-\frac{k}{D}(x^2+y^2)} \\ &= \left(\frac{k}{\pi D}\right)^{\frac{1}{2}} e^{-\frac{kx^2}{D}} \cdot \left(\frac{k}{\pi D}\right)^{\frac{1}{2}} e^{-\frac{ky^2}{D}} \\ &= \left(\frac{\pi D}{k}\right)^{-\frac{1}{2}} e^{-\frac{kx^2}{D}} \cdot \left(\frac{\pi D}{k}\right)^{-\frac{1}{2}} e^{-\frac{ky^2}{D}} \\ &= p(x) \cdot p(y), \end{aligned}$$

both of which solve the 1-dimensional F-P equations we require them to. Thus substituting expression 1.27 into equation 1.25, we see that both sides are equivalent to 0. This shows that in 2 dimensions we should again expect to see a steady state Gaussian distribution of the quantity being described by the process, in our case velocity, with density

$$p(x,y) = \frac{k}{\pi D} e^{-\frac{k}{D}(x^2+y^2)}.$$

It is easy enough to extend this idea to 3 dimensions too. We would have the 3-dimensional F-P equation

$$\frac{\partial p}{\partial t} = \frac{\partial (kxp)}{\partial x} + \frac{\partial (kyp)}{\partial y} + \frac{\partial (kzp)}{\partial z} + \frac{1}{2} D \left(\frac{\partial^2 p}{\partial x^2} + \frac{\partial^2 p}{\partial y^2} + \frac{\partial^2 p}{\partial z^2} \right),$$

which when looking for a steady state solution becomes

$$\frac{\partial(kxp)}{\partial x} + \frac{\partial(kyp)}{\partial y} + \frac{\partial(kzp)}{\partial z} + \frac{1}{2}D\left(\frac{\partial^2 p}{\partial x^2} + \frac{\partial^2 p}{\partial y^2} + \frac{\partial^2 p}{\partial z^2}\right) = 0. \tag{1.28}$$

Again using the general form for the n-variate Gaussian distribution, given in equation 1.26, with $n = 3$, we have the proposed solution

$$p(x, y, z) = \left(\frac{\pi D}{k}\right)^{-\frac{3}{2}} e^{-\frac{k}{D}(x^2+y^2+z^2)}, \tag{1.29}$$

again noting that x, y and z are independent. This can again be split into three univariate Gaussian distributions

$$\begin{aligned} &= \left(\frac{\pi D}{k}\right)^{-\frac{1}{2}} e^{-\frac{kx^2}{D}} \cdot \left(\frac{\pi D}{k}\right)^{-\frac{1}{2}} e^{-\frac{ky^2}{D}} \cdot \left(\frac{\pi D}{k}\right)^{-\frac{1}{2}} e^{-\frac{kz^2}{D}} \\ &= p(x) \cdot p(y) \cdot p(z), \end{aligned}$$

which each solve the 1-dimensional F-P equation and thus $p(x, y, z)$ is indeed the solution to the 3-dimensional F-P equation 1.28, confirming that we should expect a steady state Gaussian distribution of velocities in all 3 dimensions when applying the PRW model.

Speed distributions in 1, 2 and 3 dimensions

For our simulations and subsequent analysis, we consider the distribution of speeds rather than velocities. The velocities in each direction are independent and Gaussian with zero mean, thus the sum of squared normalized velocities follows a chi-square distribution on n degrees of freedom in n dimensions. The speed u is the square root of the sum of the velocity components, meaning in n dimensions u will follow the distribution from the chi family with n degrees of freedom (Weisstein, E. W., 2004). This takes different forms in different dimensions.

In 1 dimension this is the half-normal distribution with density function

$$f(u; \sigma) = \frac{\sqrt{2}}{\sigma\sqrt{\pi}} e^{-\frac{u^2}{2\sigma^2}},$$

for scale parameter σ and speed u; in our case $\sigma = S$.

In 2 dimensions speeds should follow the Rayleigh distribution with density

$$f(u; \sigma) = \frac{u}{\sigma^2} e^{-\frac{u^2}{2\sigma^2}},$$

for scale parameter σ and speed u; in our case $\sigma = S/\sqrt{2}$ and in 3 dimensions, the Maxwell-Boltzmann distribution with density

$$f(u;\sigma) = \sqrt{\frac{2}{\pi}}\frac{u^2}{\sigma^3}e^{-\frac{u^2}{2\sigma^2}},$$

for scale parameter σ and speed u; in our case $\sigma = S/\sqrt{3}$. It is briefly demonstrated here how to go from the velocity distribution to the speed distribution, in 3D.

The general form of the Maxwell-Boltzmann distribution of speeds, which gives the probability per unit speed of finding the given cell at speed near s is

$$f(s) = \left(\frac{m}{2\pi kT}\right)^{\frac{3}{2}} 4\pi s^2 e^{-\frac{ms^2}{2kT}},$$

where m is the mass of the cell, k is the Maxwell-Boltzmann constant $k = 1.3806 \times 10^{-23}$ and T is the thermodynamic temperature. The general form of the velocity distribution in 3 dimensions is given by

$$f(v_x, v_y, v_z) = \left(\frac{m}{2\pi kT}\right)^{\frac{3}{2}} e^{-\frac{m(v_x^2+v_y^2+v_z^2)}{2kT}},$$

for the three components of velocity v_x, v_y, v_z.

Comparing this to the Gaussian velocity distribution derived above for 3 dimensions, equation 1.29, we see that if the velocity components are x, y, z then $k/D = m/2kT$. Using the SDE for the 3-dimensional PRW model,

$$d\mathbf{v} = -\frac{1}{P}\mathbf{v}dt + \sqrt{\frac{2S^2}{3P}}d\mathbf{W}(t),$$

with diffusion coefficient $D_3 = S^2P/3$, where $k = 1/P$ and $D = 2S^2/3P$, we can see that $m/2kT = 3/2S^2$. Thus we can substitute this into the general form of the Maxwell-Boltzmann speed distribution to obtain the distribution we would expect for speeds amongst cells that follow the PRW model in 3 dimensions. Doing so we obtain

$$f(u;S) = \left(\frac{3}{2\pi S^2}\right)^{\frac{3}{2}} 4\pi u^2 e^{-\frac{3u^2}{2S^2}},$$

which is equivalent to the density in equation 1.8.

Chapter 2

A framework for modelling cell motility based on the Persistent Random Walk model: Part II - Methodology and Application

2.1 Using the PRW model to describe cell motility in 3 dimensions

2.1.1 *In silico* tests

In order to use the PRW model to describe motility in 3D, we have created a workflow to rigorously assess the fit of the PRW model to cell tracking data by using the data set to parametrize the model before verifying the fit using the statistical measures outlined in the previous chapter. This framework involves: inputting formatted cell tracking data, estimating S and P, and verifying model fit using additional statistical measures. A diagram of the workflow is provided for clarity in figure 2.1. Validation of the framework is important to ensure our method extracts the correct parameters S and P; it also allows us to assess if the model is appropriate for the data. The code which runs the framework in 3D

along with all necessary functions is available online via the link in Appendix A.

In order to validate the workflow, we used *in silico* data generated from the SDE

$$d\mathbf{v} = -\frac{1}{P}\mathbf{v}\,dt + \sqrt{\frac{2S^2}{3P}}\,d\mathbf{W}(t), \tag{2.1}$$

with specified values of S and P. This allowed us to create a data set similar to the experimental set and conduct the validation tests with prior knowledge of the parameters. Refinement of the method was then carried out until the estimates were sufficiently accurate.

Cell tracking data entered into the framework must be an array outlining the positions and velocities of each cell at each time point. If only positions within tracks are available, as will be seen in the experimental data sets, the velocity of each cell at each time point is estimated from the difference in the current and previous position divided by the time step. For the *in silico* data sets we numerically simulate equation 2.1 along with $d\mathbf{x}/dt = \mathbf{v}$ using the `simByEuler` function (MATLAB, Financial Toolbox, Mathworks (2019)) to simultaneously obtain both cell positions and velocities in the data set.

In addition to S and P, it is necessary to specify the numerical time step, dt, the total time of the simulation, $dt \times$ Nperiods, with Nperiods being the number of simulation periods, the number of cells N, and the initial position and velocity vectors $\mathbf{x}_0$ and $\mathbf{v}_0$ for all cells. Figures 2.2a) and b) show sample plots of the 3D tracks generated by the workflow.

2.1.2 S estimate

Parameter S is defined as the root mean squared speed of cells once the system reaches steady state. The root mean squared speed ($RMSS$) at time t across all cells is calculated in 3D as

$$RMSS(t) = \sqrt{\langle v_x(t)^2 + v_y(t)^2 + v_z(t)^2 \rangle}, \tag{2.2}$$

where the average $\langle . \rangle$ is over all cells, and the 3D components of the velocity at time t are given by $v_x(t)$, $v_y(t)$ and $v_z(t)$. We take the average of $RMSS(t)$ at

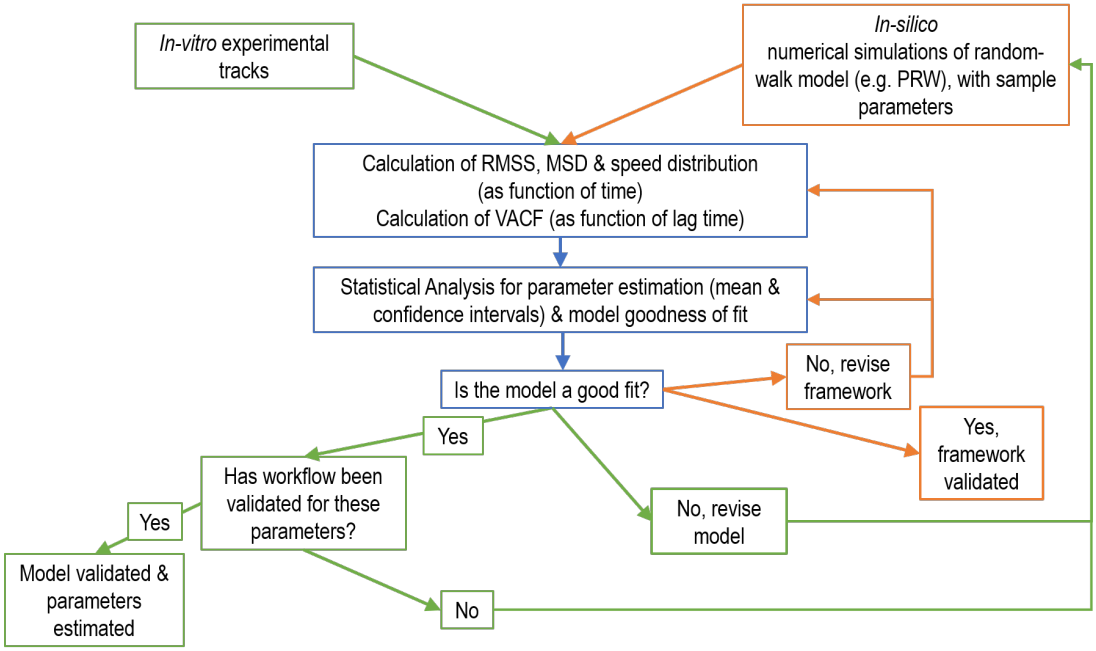

Figure 2.1: **Diagram of the workflow.** An overview of the flow of the framework described. Code to carry out this analysis can be found online via the links in Appendix A.

all times to obtain an overall estimate of S, $\hat{S}$,

$$\hat{S} = \frac{1}{T}\sum_{t=0}^{T} RMSS(t) = \frac{1}{T}\sum_{t=0}^{T}\sqrt{\langle v_x(t)^2 + v_y(t)^2 + v_z(t)^2\rangle}, \tag{2.3}$$

as it is assumed that experimental data would initially be at steady state.

The framework outputs a plot of the $RMSS$ time series from which an estimate of S is obtained, and a histogram of the speed distribution at specified time points with the corresponding Maxwell-Boltzmann density function with estimated parameter $\hat{S}$ overlaid. This is depicted in figure 2.3 for a simulation with $S = 1$ and $P = 1$. In this simulation, all cells had initial speed of 1, allowing us to obtain the stationary speed distribution more rapidly. Plots c(i)-(iv) demonstrate how the speed distribution of cells settles to the stationary distribution.

We will also look at a confidence interval for the S estimate. The calculated $RMSS$ values observed at time step $\tau = 1, 2, \ldots,$ Nperiods, taken in sequence, are modelled as an autoregressive time series of order 1, denoted AR(1). This

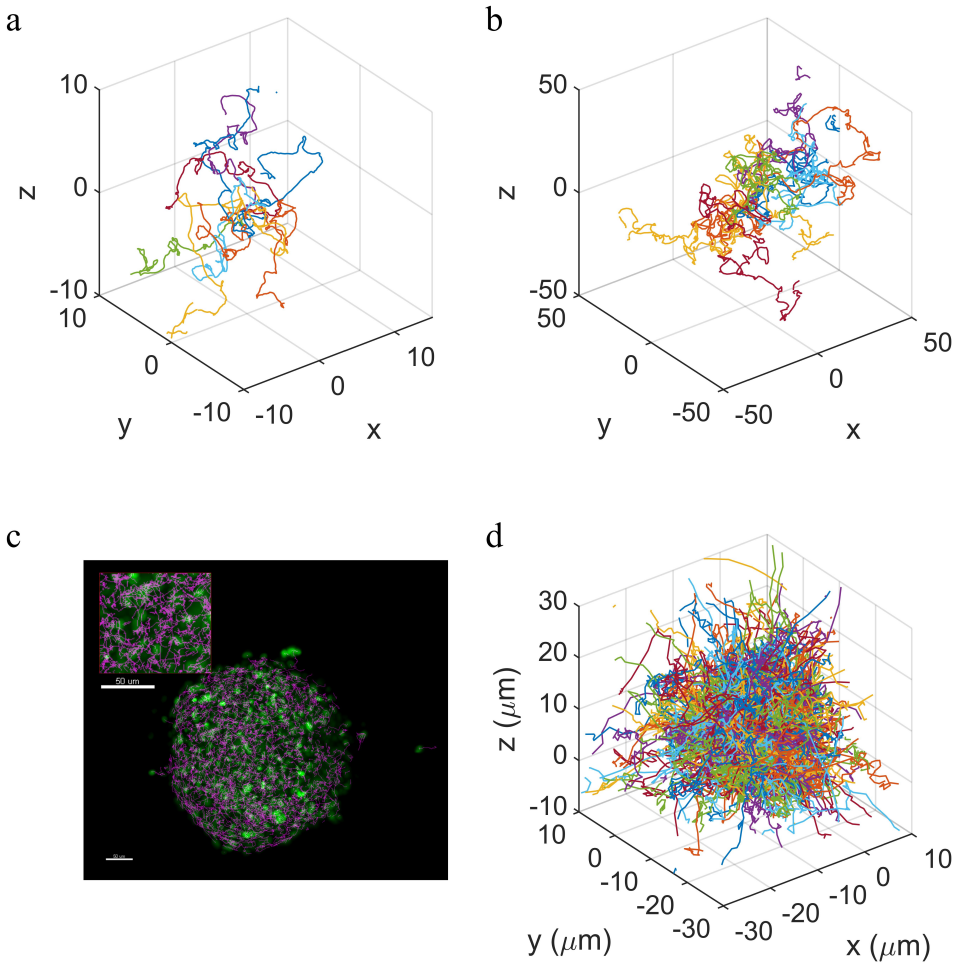

Figure 2.2: **Examples of cell trajectories in 3D. a)** *in silico* data with parameters $S = 1$, $P = 1$, $dt = 0.05$ and NPeriods $= 1000$. Cells are initialised at the origin, $\mathbf{x}_0 = \mathbf{0}$, with speed S and orientation sampled uniformly from the unit sphere. Plot shows tracks from 10 cells as example trajectories. **b)** *in silico* data with parameters $S = 25$, $P = 0.1$, $dt = 0.05$ and NPeriods $= 480$. Initial positions and velocities as in a). Plot shows tracks from 10 cells as example trajectories. **c)** Experimental *in vitro* images with green indicating location of cell nuclei, and purple the overlay of cell tracks identified using tracking software, from Richards *et al.* (2018). Inset of zoomed in tracks and scalebar. **d)** The corresponding experimental trajectories from c) plotted within the framework. Initial positions and velocities taken from first entries for each track.

means the current value of the process depends on the past only through the value of the process in the previous time step.

A time series Y_τ is called autoregressive of order 1 if it satisfies

$$Y_\tau = \delta + \theta Y_{\tau-1} + \epsilon_\tau,\ \epsilon_\tau \sim^{i.i.d} \mathrm{N}(0, \sigma^2),$$

where δ and θ are constants and the random errors ϵ_τ, assumed to be independent and identically distributed, are modelled as a normal random variable with mean 0 and finite variance σ^2 (Brockwell & Davis, 2016). The consecutive observations for such a time series are clearly dependent and their correlation is equal to θ. Because of this dependence, the standard confidence intervals for the mean cannot be used with $RMSS$ and an adjustment to the sample size is needed.

Using the fact that the mean of this time series is S, we construct a 95% confidence interval for $\hat{S}$. We use the sample size adjustment from Zwiers & von Storch (1995), the effective sample size, calculated as

$$n_e = \frac{\mathrm{Nperiods}}{1 + 2\sum_{\tau=1}^{\mathrm{Nperiods}-1}\left(1 - \frac{\tau}{\mathrm{Nperiods}}\right)\rho_1^\tau}, \tag{2.4}$$

where Nperiods is the number of observations in the $RMSS$ time series and ρ_1 is the lag-1 correlation coefficient obtained using the `autocorr` function (MATLAB, Econometrics Toolbox, (Mathworks, 2019)) which calculates the sample autocorrelation coefficient for the time series.

We use the following formulae to obtain a 95% confidence interval for $\hat{S}$. For $n_e > 30$ we can assume normality and calculate the interval using

$$\left[\hat{S} \pm Z(0.025)\frac{s}{\sqrt{n_e}}\right], \tag{2.5}$$

where s is the sample standard deviation of the $RMSS$ values, $Z(0.025)$ is the critical value of the cumulative normal distribution at 0.975 and n_e is the equivalent sample size as above.

When $n_e \leq 30$ we must use the t-distribution with $n_e - 1$ degrees of freedom, thus the interval here is calculated using

$$\left[\hat{S} \pm t_{n_e-1}(0.025)\frac{s}{\sqrt{n_e}}\right]. \tag{2.6}$$

An example 95% confidence interval for $\hat{S}$ from the *in silico* data in figure 2.3 where $S = 1$ and $\hat{S} = 0.9973$ is $[0.9937, 1.0010]$.

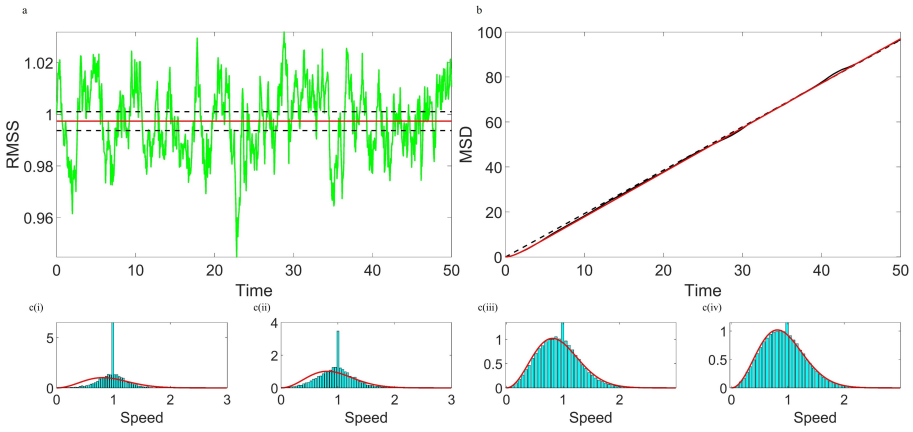

Figure 2.3: **Example *in silico* output from the framework in 3D** for 1000 cells with NPeriods = 1000, $dt = 0.05$ and $S = P = 1$. Cells are initialised at the origin, $\mathbf{x}_0 = \mathbf{0}$, with speed S and orientation sampled uniformly from the unit sphere. **a)** $RMSS$ over time (green line) with estimated speed $\hat{S} = 0.9973$ (red line) 95% confidence interval $[0.9937, 1.0010]$ ($n_e = 51$) (black dashed lines). **b)** Calculated MSD vs time (black line) with model predicted MSD (red line) and a straight line fitted to the calculated MSD (black dashed line), enabling S and P estimates to be verified through the gradient of the line being equated to $2S^2P$. The inferred P estimate here is $\hat{P} = 0.9806$, whilst the framework estimated value is $\hat{P} = 0.9951$ with 95% confidence interval [0.9232, 1.0791]. **c)** Histograms of cell speed distributions at $t =$ **i** 0.25, **ii** 0.5, **iii** 2.5 and **iv** 25, and Maxwell-Boltzmann distribution (red curve) with estimated parameter $\hat{S} = 0.9973$ overlaid.

2.1.3 P estimate

The $VACF$ is used to estimate P. This is done by first calculating the $VACF$ from 3D data using

$$VACF(t) = \langle (v_x(0) \cdot v_x(t)) + (v_y(0) \cdot v_y(t)) + (v_z(0) \cdot v_z(t)) \rangle \,, \qquad (2.7)$$

where the average $\langle . \rangle$ is over all cells. From equation 1.6, we can estimate $-1/P$ as the slope of a plot of $\ln(VACF)$ against t. We note that we are using a special case of the OU process in which correlations in the increments of v_x, v_y and v_z are absent, simplifying the $VACF$ calculation and thus potentially affecting the model's ability to describe any data sets where these correlations may be present.

To obtain an estimate of the gradient of this line, we consider a simple linear regression model fitted to the observed $\ln(VACF)$ values and in doing so directly calculate the estimate for P. We here utilise the fact that parameter P is estimated as the negative reciprocal of the slope of a linear regression model, call it b, fitted to $\ln(VACF)$, i.e. $\hat{P} = -1/b$. We obtain point estimates for intercept and slope coefficients, along with standard error estimates for these values. Denoting the intercept by a and the slope coefficient by b, we then use the fitted line $Y = a + bt$ to assess mean squared error and choose a suitable cut-off value for our set of $\ln(VACF)$ values. The line corresponding to the chosen cut-off is then plotted onto the $\ln(VACF)$ vs t plot to demonstrate the subset of values used in estimating P.

Given that our observations are serially correlated, and thus the errors involved in fitting this regression line will also be correlated, we fit this line using feasible generalised least squares (FGLS) instead of the traditional ordinary least squares (OLS) method.

OLS assumes homoscedasticity of errors and due to the inherent correlation in our data, we have dependent errors. For correlated errors with known correlation matrix Ω, Generalised Least Squares (GLS) can be used instead. In this method the regression equation is multiplied by $\Omega^{-1/2}$ and consequently, generalised least squares estimators of the regression coefficients are found by minimizing the squared Mahalanobis distance of the vector of residuals.

The added complication is that we also don't know the structure of Ω, and so we must resort to FGLS which finds the estimators of the regression coefficients $\hat{\beta}$ using OLS and uses the innovations from this model to estimate $\hat{\Omega}$.

The model with this $\hat{\Omega}$ is then fitted using GLS and the innovations from this fit are then used to estimate $\hat{\Omega}$ again. This is repeated until there is convergence in $\hat{\beta}$ and $\hat{\Omega}$. As such we use the `fgls` function (MATLAB, Econometrics Toolbox, (Mathworks, 2019)) to obtain the line of best fit along with estimates for the slope coefficient and its corresponding standard error estimate.

As $VACF$ tends towards zero there is increasing noise in the estimate of $\ln(VACF)$, and an estimate of P that uses all of this data would be erroneous. Figure 2.4, particularly plot a(i), shows just how noisy the data can be. To ensure the estimates are not affected by this noise, we only fit our regression model to a subset of data points by implementing a cut-off value. Observations of $\ln(VACF)$ falling below this value are excluded from the data set.

To determine this new subset, we systematically try a range of cut-off values for $\ln(VACF)$, the line being fitted only to those values above the cut-off, by defining a cut-off test vector with equally spaced entries between the minimum and maximum values of $\ln(VACF)$. We subsequently calculate the mean squared error (MSE) for each fit and choose the cut-off for which the subset includes the most data points such that MSE < 0.5, and proceed as above using the `fgls` function to carry out the rest of the analysis.

This choice of MSE cut-off will depend on the simulation parameters, for example number of cells N, the required accuracy of parameter estimates and MSE obtained from fitted models, but the methodology provides a repeatable and adjustable method for estimating P and the cut-off is one of the parameters that is easily changed. We also restrict the search to subsets with more than 5 data points to allow FGLS to be used, as we are fitting 4 parameters in the regression model (intercept, slope, variance and autocorrelation).

To obtain an estimate $\hat{P}$, we apply the `fgls` function to the resulting subset of $\ln(VACF)$, choosing to fit to $-\ln(VACF)$ to simplify the algebra and make $\hat{P} = 1/\hat{\beta}$, where $\hat{\beta}$ is the estimated slope coefficient.

To obtain a 95% confidence interval for $\hat{P}$, we assume that both $\hat{P}$ and $\hat{\beta}$ are positive quantities, a realistic assumption since we cannot obtain a negative persistence time.

Now, the upper bound of a 95% confidence interval for $\hat{P}$ is some value P_U such that

$$\begin{aligned} 0.025 &= Pr(\hat{P} > P_U) \\ &= Pr\left(\frac{1}{\hat{\beta}} > P_U\right) \\ &= Pr\left(\hat{\beta} < \frac{1}{P_U}\right). \end{aligned}$$

The lower bound of a 95% confidence interval for the slope $\hat{\beta}$ is $\hat{\beta}_L$ such that $0.025 = Pr(\hat{\beta}_L > \hat{\beta})$, and so $\frac{1}{P_U} = \hat{\beta}_L$ and $P_U = \frac{1}{\hat{\beta}_L}$.

Similarly, the lower bound of a 95% confidence interval for $\hat{P}$ is P_L such that

$$\begin{aligned} 0.025 &= Pr(\hat{P} < P_L) \\ &= Pr\left(\frac{1}{\hat{\beta}} < P_L\right) \\ &= Pr\left(\hat{\beta} > \frac{1}{P_L}\right). \end{aligned}$$

The upper bound of a 95% confidence interval for the slope b is b_U such that $0.025 = Pr(\hat{\beta}_U < \hat{\beta})$, and so $\frac{1}{P_L} = \hat{\beta}_U$ and $P_L = \frac{1}{\hat{\beta}_U}$.

We can then obtain a confidence interval for our P estimate by building the 95% confidence interval for slope coefficient $\hat{\beta}$ as

$$\left[\hat{\beta}_L = \hat{\beta} - t_{n-2}(1-\alpha/2)\ SE_{\hat{\beta}},\ \hat{\beta}_U = \hat{\beta} + t_{n-2}(1-\alpha/2)\ SE_{\hat{\beta}}\right] \tag{2.8}$$

where n is the number of data points in the subset, $\alpha = 0.05$, t_{n-2} denotes the t-distribution with $n-2$ degrees of freedom and $SE_{\hat{\beta}}$ is the estimated standard error of $\hat{\beta}$, and transforming this to obtain the 95% interval for $\hat{P}$ as

$$\left[\frac{1}{\hat{\beta}_U}, \frac{1}{\hat{\beta}_L}\right]. \tag{2.9}$$

The plots in figure 2.4 are formed from fitting the regression model to $\ln(VACF)$, and show how different $\ln(VACF)$ data sets force subsets of this data of different lengths to be used for FGLS fitting and subsequent P estimation, according to the MSE cut-off algorithm explained above.

We expect this regression line to have an intercept, which is also fitted in the model, at $\ln(S^2)$, and so it can be useful to compare this value to the estimated intercept given by the `regress` function (MATLAB Statistics and Machine Learning Toolbox, (Mathworks, 2019)) as another way of assessing how well the PRW model can explain a data set.

Figure 2.4 shows examples of the framework output $\ln(VACF)$ plots for $S = 1$, $P = 1$ and 10 and where dt is taken to be 0.01, 0.1 and 1, and the choice of cut-off is determined by the above algorithm. This produces P estimates, along with their 95% confidence intervals, which can be seen in table 2.1.

Numerical simulations of stochastic differential equations, and associated statistical measures, are strictly valid in the limit as $dt \to 0$. In our simulations, when the persistence time, P, is comparable to dt, we see the reduction in predictive power; for example when $dt = P = 1$, as in figure 2.4a(iii), the confidence interval doesn't include what we know to be the true value of P.

2.1.4 Mean Squared Displacement

Upon calculating estimates for both S and P, the theoretical MSD from equation (1.7) can be compared with the calculation from the data:

$$MSD(t) = \left\langle (x(t) - x(0))^2 + (y(t) - y(0))^2 + (z(t) - z(0))^2 \right\rangle, \tag{2.10}$$

where the average $\langle . \rangle$ is over all cells and the position vector at time t is given by $(x(t), y(t), z(t))$. Figure 2.3b) shows a plot of the calculated MSD vs model MSD for $S = 1, P = 1$ as an example.

We note from equation 1.7 that in the limit as $t \to \infty$, the expression for MSD becomes $MSD(t) = 2S^2Pt$, the equation of a straight line with a slope of $2S^2P$. Fitting a regression model to the calculated MSD vs t plot, making use of FGLS since the MSD observations from each time step will depend on previous MSD observations, we can also infer $\hat{P}$ using $\hat{S}$ as

$$\hat{P} = \frac{\text{slope}_{\text{MSD}}}{2\hat{S}^2},$$

with $\text{slope}_{\text{MSD}}$ being the estimated slope coefficient from the FGLS fit to the

MSD vs t plot. In doing so for data shown in figure 2.4, the inferred P estimates as in table 2.1 were obtained.

We posit that the total simulation time for the data set in figure 2.4b(i) is not large enough compared to $P = 10$, to use the fact that $MSD(t) \to 2S^2Pt$ as $t \to \infty$ to justify a linear fit to the data, hence the very poor estimate of P found through this method. In this case a non-linear fit of equation 1.7 should be carried out to infer an estimate for P.

2.1.5 Discussion of model parameters and output from *in silico* simulations

Estimation of parameters from *in silico* data allows us to validate our method and assess the accuracy of our estimates. Having demonstrated our framework can successfully extract these values from 3-dimensional *in silico* simulations, we will go on to estimate S and P from experimental data in the next section, and also check that the workflow is robust for *in silico* data generated from experimental estimates for S and P.

It is clear from figure 2.3 that the simulated speeds follow the Maxwell-Boltzmann distribution after enough time has passed for the stationary distribution to be reached. This means we can be reasonably confident that the average of the $RMSS$ will be a good estimate of S in a population that follows the PRW model, and this is confirmed by the narrow confidence intervals calculated for the examples given.

We can also consider the velocity distribution for each of the components of the velocity which we assume to be Gaussian. For consistency, we conduct an Anderson-Darling test using the function `adtest` (MATLAB Statistics and Machine Learning Toolbox, (Mathworks, 2019)) on each of the components of velocity (v_x, v_y, v_z) to check that the assumption is indeed satisfied. If this assumption is violated then we wouldn't expect speeds to follow the Maxwell-Boltzmann distribution which depends on these Gaussian velocities.

The Anderson-Darling test was conducted at each time point across all cells with the final time point being taken particularly into consideration. For all

	$\mathbf{S}, \mathbf{P}, \mathbf{dt}$	$\hat{\mathbf{S}}$	$\hat{\mathbf{S}}_\mathbf{I}$	$\hat{\mathbf{P}}$	$\hat{\mathbf{P}}_\mathbf{M}$
3D	1, 1, 0.05	0.9973	0.9965	0.9951	0.9806
		[0.9937, 1.0010]		[0.9232, 1.0791]	
	1, 1, 0.01	0.9978	0.9952	0.9893	0.9141
		[0.9892, 1.0064]		[0.9473, 1.0352]	
	1, 1, 0.1	0.9993	1.0072	1.0919	0.9533
		[0.9968, 1.0018]		[0.9999, 1.2505]	
	1, 1, 1	1.0020	0.9613	1.1497	0.9845
		[1.011, 1.0030]		[1.0661, 1.2474]	
	1, 10, 0.01	1.0097	1.0101	10.3752	3.6942
		[0.9911, 1.0282]		[9.9821, 10.3752]	
	1, 10, 0.1	0.9943	0.9912	9.8853	9.4376
		[0.9899, 0.9987]		[9.5095, 10.7919]	
	1, 10, 1	1.0007	1.0005	10.3866	9.7890
		[0.9981, 1.0033]		[9.8946, 10.9032]	
	25, 0.1, 0.05	25.0458	25.3024	0.0996	0.0962
		[24.9091, 25.1825]		[0.0978, 0.1015]	
2D	1, 1, 0.1	1.0050	0.9981	1.0261	0.8827
		[0.9981, 1.0118]		[1.0088, 1.0441]	
	25, 2, 0.05	25.2132	25.0532	2.0327	1.9518
		[25.0702, 25.3562]		[1.7412, 2.4416]	

Table 2.1: Parameter estimates for all *in silico* data sets considered in the work. S, P, dt refer to the parameters used when simulating the *in silico* data, $\hat{S}_I$ is the S estimate inferred from the FGLS regression model fit to the $\ln(VACF)$ data, $\hat{P}_M$ is the P estimate inferred from the regression model fit to the MSD data. 95% confidence intervals are given where appropriate.

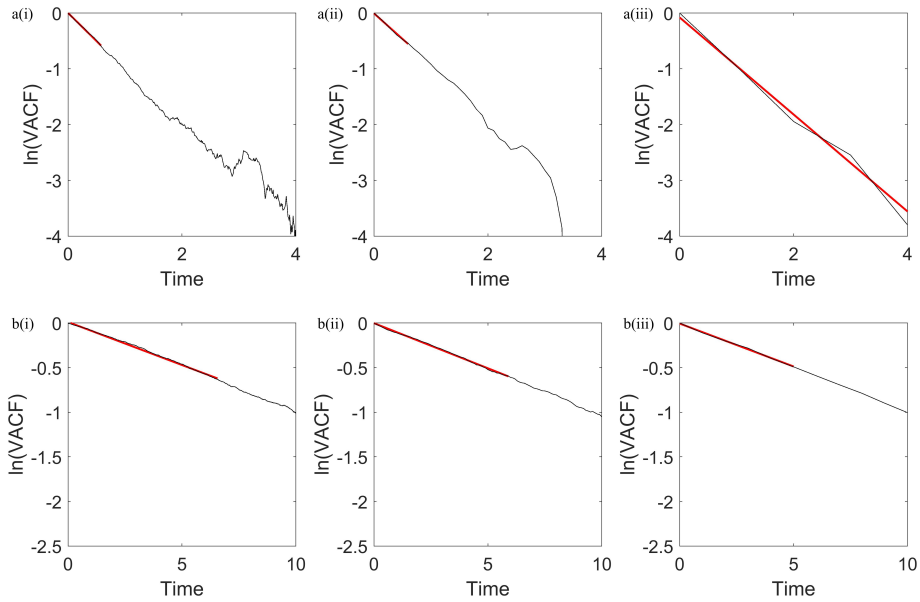

Figure 2.4: **Estimation of P using $\ln(VACF)$ with algorithmic cut-off points for line fits in the 3D framework**. Calculated values of $\ln(VACF)$ are shown (black line) with FGLS line fits (red line). FGLS line fits differ in length in each panel due to subsets of $\ln(VACF)$ data of varying length used in the estimation of $\hat{P}$, according to the MSE cut-off algorithm defined in the main text. **a)** $S = 1, P = 1$, Nperiods $= 1000$, **i** $dt = 0.01$, **ii** $dt = 0.1$ and **iii** $dt = 1$ respectively for 1000 cells. P estimates from left to right along with 95% confidence intervals are $\hat{P} = 0.9893$ [0.9473, 1.0352], $\hat{P} = 1.0919$ [0.9999, 1.2505] and $\hat{P} = 1.1497$ [1.0661, 1.2474]. **b)** $S = 1, P = 10$, Nperiods $= 1000$, **i** $dt = 0.01$, **ii** $dt = 0.1$ and **iii** $dt = 1$ respectively for 1000 cells. P estimates from left to right along with 95% confidence intervals are $\hat{P} = 10.3752$ [9.9281, 10.8644], $\hat{P} = 9.8853$ [9.5095, 10.2919] and $\hat{P} = 10.3866$ [9.8946, 10.9302].

in silico data sets, the test showed that at the final time point all components of velocity were normally distributed, hence giving further confidence in the S estimate.

When we are looking at estimating P we also need to be careful with the timescale we are simulating over. The simulation interval dt needs to be much smaller than P to be able to see the persistence in velocity over several time periods and subsequent decay of the velocity autocorrelation. We should also ensure that the total simulation time is much larger than P to be able to see the effect of the decay in correlation. We should therefore get a more accurate P estimate with values of dt much smaller than P and a high number of simulation periods.

We also note that the choice of MSE threshold is important here. When testing the framework with *in silico* data, estimates of both S and P were seen to be robust to MSE choice, even when the threshold was as small as 0.05. The MSE should not be too large, but overfitting to the *in silico* data could lead to poor prediction in experimental data sets. The MSE threshold was thus set at 0.5 to be consistent with the chosen threshold for the experimental data sets in our analyses. In practice the MSE threshold should be set based on the data set being investigated, it being sensitive to sample size. The choice will be dependent on the amount of data once observations have been removed as per the cut-off algorithm, and values of MSE that an investigator deems acceptable in relation to the context of the experimental data itself.

Figure 2.4 shows the framework output when different values of dt are used and demonstrates how P estimates vary as dt varies between 0.01 and 1. We would expect estimates to become more accurate as dt decreases, and this is seen here when $P = 1$ but not when $P = 10$, possibly due to the way that we choose a subset of data to use when estimating P. In reality the choice of dt, number of cells and the number of simulation periods may be restricted by the data from an experiment, and so consideration of how to amend the framework in these cases may be necessary.

2.1.6 Applying the framework to 3-dimensional experimental tracking data

Specifics of the experimental data

After validating the method using pre-determined parameter values for *in silico* data, we were able to reliably use it to extract parameter values from a 3D experimental data set. The cell tracking data used here was obtained from *in vitro* tumour spheroids consisting of glioblastoma cells. These spheroids were grown and imaged with a Light Sheet Fluorescent microscope, as described in Richards *et al.* (2018), and of importance here is the fact that images were collected every 3 minutes over a 24 hour period, meaning there are 480 periods of 0.05 hours in the data set. Though the spheroids were in some instances treated with drugs, the 3 data sets we use are all controls.

The data is in the form of individual cell tracks, there being a velocity at each time step for each cell, as required. Plots of the tracks from one of these control spheroids are shown in figure 2.2d) and compared to the experimental image in 2.2 c). There were 3780, 3861 and 3808 cells in each of the three experimental data sets though only cell tracks that are recorded as starting at time 0 are included in the analysis, thus we analyse the 549, 929 and 1054 cells with such tracks for 149, 93 and 78 periods of 0.05 hours in control spheroid data sets 1, 2 and 3 respectively. This means we look at time periods of 7.5, 4.7 and 3.9 hours. Code used to format the data is available via the link given in Appendix A.

Parameter estimation and goodness-of-fit for the experimental data

Upon running the data through our framework we were able to obtain estimates for parameters S and P along with 95% confidence intervals, as stated in table 2.2. In the calculations of the confidence intervals for $\hat{S}$ we found effective sample sizes of $n_e = 16.6, 19.5$ and 29.5, resulting from sample autocorrelations of 0.8064, 0.6635, and 0.4617 at lag 1. Output plots from the framework can be seen in figure 2.5 for each of the three spheroids.

Our speed estimates agree well with the estimate of 27 μm/h obtained from

	Data set	$\hat{\mathbf{S}}$ (μm/h)	$\hat{\mathbf{S}}_\mathbf{I}$ (μm/h)	$\hat{\mathbf{P}}$ (h)	$\hat{\mathbf{P}}_\mathbf{M}$ (h)
3D	Spheroid 1	27.3137	42.4743	0.0863	0.0940
		[25.2892, 29.3382]		[0.0697, 0.1130]	
	Spheroid 2	26.9272	34.6865	0.0789	0.1289
		[25.9613, 27.8930]		[0.0677, 0.0946]	
	Spheroid 3	28.0600	35.1386	0.0976	0.1017
		[27.3979, 28.7222]		[0.0804, 0.1241]	
2D	Spheroid	2.5928	0.1064	-17.6078	0.00024
		[0.5426, 4.6430]		[-54.8113, -10.4886]	

Table 2.2: Parameter estimates for all experimental data sets considered in the work. $\hat{S}_I$ is the S estimate inferred from the FGLS regression model fit to the $\ln(VACF)$ data, $\hat{P}_M$ is the P estimate inferred from the regression model fit to the MSD data. 95% confidence intervals are given where appropriate.

the same data set for cells located inside the spheroid boundary in Richards *et al.* (2018). In terms of our estimates for P, there are very few sources in the literature which predict persistence time for any type of cell, less so for GBM cells. We note Stein *et al.* (2007) carried out similar analysis to ours studying GBM U87 cells from 2D projections of 3D images and obtained a value of $\beta = 9.3/\text{h}$, corresponding to $P = 1/\beta = 0.1075\,\text{h}$ which is similar to the values we find.

Our estimates for both S and P are additionally very consistent across the controls, making them fairly reliable for this experiment. We can also again infer P from the MSD calculations for comparison, infer S from the regression line fitted to $\ln(VACF)$, both of which are displayed in table 2.2.

Compared to the estimates from the model framework obtained through $RMSS$, independently of P, the predictions from the regression overestimate in each case, though the experimental values are also above the values that the framework estimates. We suggest that the most reliable method of estimating S is still the one using the $RMSS$ as this encompasses the definition of parameter S and provides the estimates closest to those found by experimentalists.

As further validation that our framework should be able to correctly extract

parameters from the data, figure 2.6 shows the framework output for a realistic set of parameter values as informed by running the experimental data through the framework ($S = 25$, $P = 0.1$, $dt = 0.05$, NPeriods = 100, 550 cells). This shows that our framework is still capable of estimating S and P accurately when the experimental parameter values are used, even with the restricted dt value. The estimated value of P is 0.0996 with 95% confidence interval [0.0978, 0.1015], which includes the true value of $P = 0.1$. This is a good sign we can be reasonably confident in our intervals for $\hat{P}$ that come from the experimental data P estimates.

By conducting this analysis we are able to explore 'realistic' parameters in the framework and see how well it is capable of estimating parameters of this magnitude. This enables us to see if we can indeed make accurate predictions about the experimental data using the framework, but also to test the robustness of it when parameters take values similar to these. For example we have estimated P to be around 0.1, and since we need $dt << P$ to see persistence over several time intervals, we should determine whether the framework can handle values of P which are quite close to dt, as in the experimental case where dt is 0.05.

We see from the output of this *in silico* simulation with experimental parameters that the framework is capable of handling such parameters, and thus can go on to make conclusions about the experimental data knowing that any discrepancies arising are not down to the framework's estimation capabilities, but to experimental errors or biological phenomena.

Probing model assumptions using the framework applied to experimental data

We shall now highlight some fundamental differences depicted in figure 2.5 between model predictions and experimental observations, and where possible propose rigorous statistical tests to examine whether the PRW model should be rejected. By comparing figure 2.6, which shows a very good fit of model to *in silico* data, to figure 2.5, we have confidence that differences between model pre-

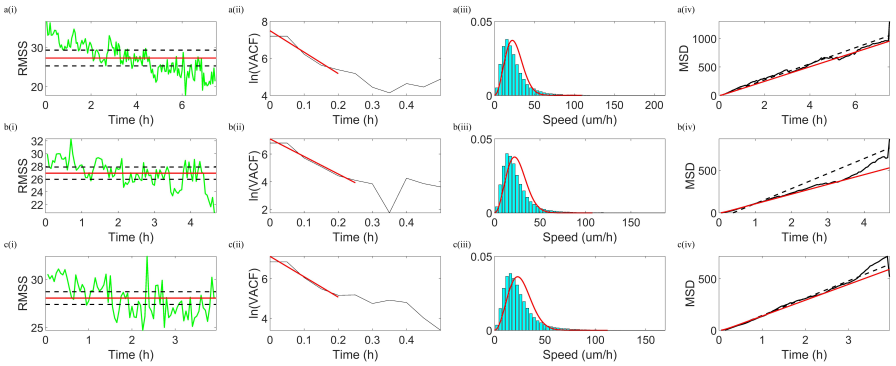

Figure 2.5: **Framework outputs for 3D experimental spheroids**, data from Richards *et al.* (2018). **a)** Spheroid 1, **b)** Spheroid 2, **c)** Spheroid 3. **i** $RMSS$ vs time (green line) with estimated average speed (red line) and 95% confidence intervals (black dashed lines). **ii** Velocity autocorrelation, calculated $\ln(VACF)$ (black line) and FGLS fit (straight red line). **iii** Histogram of speeds with Maxwell-Boltzmann density with parameter S overlaid (red curve). **iv** Calculated MSD vs time plot (black line) and model predicted MSD (red line) with line fit (black dashed line). Each row corresponds to an independent control spheroid.

dictions and observations are not due to sample size or parameter values in this case, but instead that perhaps the PRW model is not sufficient to describe this data.

To undertake statistical tests to determine whether the PRW model should be rejected, we choose to consider in more detail a subset of cell tracks which last the full length of the experiment. Firstly, we see that the Maxwell-Boltzmann distribution appears unable to completely explain the speed distribution data. If we consider only the final cell speeds in each of these tracks (figures 2.7a(iii), b(iii) and c(iii)), we have a set of independent speeds which should follow the Maxwell-Boltzmann distribution, as we are looking at a fairly large number of cells (76, 71 and 56) over a long time (149, 93 and 78 periods respectively).

We can first conduct the Anderson-Darling test on the velocities in the experimental data sets and upon doing so, even if we restrict the test to just the full length tracks in each data set, we are still led to reject the null hypothesis in all cases. This suggests that the velocities are not normally distributed and so consequently we shouldn't expect the speeds to follow the Maxwell-Boltzmann distribution.

Further, carrying out a Kolmogorov-Smirnov test (MATLAB `kstest`, Statistics and Machine Learning Toolbox, (Mathworks, 2019)) on the final cell speeds of full length tracks for each control spheroid with parameter $\hat{S}$ as estimated from the data through our framework, we see that in all cases this test instructs us to reject the null hypothesis that the data follows the Maxwell-Boltzmann distribution. Furthermore, we see from figures 2.7a(iv), b(iv) and c(iv) that the mean speeds of each cell with a full length track are not clustered around the mean of the expected Maxwell-Boltzmann distribution based on the estimated speed parameter $\hat{S}$. This leads us to believe that each cell monitored over the full experiment isn't itself displaying speeds following the Maxwell-Boltzmann distribution with this parameter $\hat{S}$.

All of this suggests that the cell speeds are not what we would expect if the cells behaved as per the model, and so there are some cells travelling quite a bit faster and some cells quite a lot slower than the estimated mean speed (estimated

mean speed ± standard deviation, spheroid 1: $27.3137 \pm 0.0027\,\mu$m/h, spheroid 2: $26.9272 \pm 0.0028\,\mu$m/h, spheroid 3: $28.0600 \pm 0.0026\,\mu$m/h).

This provokes interesting biological questions about why some cells are able to travel at higher speeds than their counterparts and perhaps looking at where these cells lie in the spheroid would provide some insight into this difference and differences in motility mechanisms across cells.

Upon plotting individual cell speeds across the experiment, we see from figure 2.8 that there are indeed some cells with abnormally high speeds at certain times, and the peaks in speed are coming from the same cells, generally those with higher mean speeds overall, though their speed is not consistently higher than we would expect.

We could probe this more by looking more in detail at how the speed distributions vary over time, monitoring when this shift in the peak of the distribution happens and when the high-speed outliers become so, to determine whether these mean speeds are so high due to extreme values at later times, or are simply down to chance.

Secondly, in figure 2.5, we see that the $RMSS$ appears to be a function of time, with the data suggesting a linear decrease, in conflict with the underlying assumptions of the PRW model. We questioned whether this trend for decreasing $RMSS$ over time is due to the decreasing number of tracks involved in the calculation as time goes on, due to initial filtering of the data according to the start time of a track. However $RMSS$ plots created with only the full length tracks as used in figure 2.7 show a similar downward trend (data not shown) and thus more data is needed to investigate this changing of speed with time. We are also assuming here that the system is already in steady state due to the cells being grown for 3 days before the tracking started and time 0 is 3 hours after the spheroid had been placed in the microscope. This assumption could be wrong and could explain the decrease in speed over the time interval we are considering. Nevertheless, it is clear that the $RMSS$ time series plot is one of the first indicators from the framework of whether a data set has a constant average speed, and thus one of the first ways to assess the suitability of the PRW model

for describing a data set.

Thirdly, in figure 2.5, we see that for all of the control spheroids, the model MSD underpredicts the calculated MSD, leading us to take care with the P estimates inferred from the MSD calculations. This underprediction agrees with the previously observed superdiffusive nature of cells in 3D (Yurchenko *et al.*, 2019; Luzhansky *et al.*, 2018; Takagi *et al.*, 2008).

Finally, in figure 2.5, the plots of $\ln(VACF)$ against time, for which the PRW model predicts to be linear with slope $-1/P$, show a reasonable fit. Moreover the noise in the data leads the method for estimating P to only use a small subset of data points when fitting the straight lines, possibly affecting the reliability and accuracy of our estimates. The nonlinear nature of the $\ln(VACF)$ plots suggests there may be some biological factors which stop the model from being able to accurately estimate P from these plots.

In this framework we are assuming that all cells are identical and independent, which is intuitively unrealistic, and accounting for possible differences in persistence time between cells could enable us to better estimate what these parameter values may be. It has been suggested in the literature (Wu *et al.*, 2014; Yurchenko *et al.*, 2019; Takagi *et al.*, 2008) that populations of cells may have several subpopulations with different persistence times. We don't explore this idea here, but one could change the framework accordingly to account for this by using a different governing model that allows for heterogeneity in individual values of population parameters. The statistical measures calculated in the framework currently based on the PRW model could not be adjusted sufficiently to account for significant heterogeneity in the population, the model itself assuming that there is one S and one P value for the entire population.

To illustrate this we ran heterogeneous *in silico* data sets through the framework which consisted of 2 possible S or P values. S values were chosen based on some cells reaching speeds of 50μm/h, to try and probe the effect of having a small number of cells with a very different average speed to the rest. P values used were based on experimental estimates being close to 0.1 and an arbitrary upper limit of 10 hours which is unlikely to be reached by most cells. The data

S	**P**	**Ratio**	$\hat{\mathbf{S}}$	$\hat{\mathbf{P}}$
50, 1	1	10:90	15.0918 [14.2269, 15.9567]	-1.4801 [-4.0881, -0.9036]
50, 1	1	50:50	34.8237 [34.2203, 35.4270]	-12.5562 [0.7457, -0.6665]
50, 1	1	90:10	48.1803 [46.5970, 49.7637]	0.5649 [0.4528, 0.7507]
1	10, 0.1	10:90	0.9982 [0.9907, 1.0058]	0.4803 [0.2655, 2.5136]
1	10, 0.1	50:50	1.0035 [0.9970, 1.0100]	14.0923 [11.8202, 17.4460]
1	10, 0.1	90:10	0.9988 [0.9912, 1.0065]	11.4860 [10.6100, 12.5197]

Table 2.3: Parameter estimates using the framework for heterogeneous *in silico* data sets. True values of S and P used in the populations are given and the proportions of cells with those true parameter values are indicated in the 'Ratio' column. 95% confidence intervals are given where appropriate.

sets used and results gained are shown in table 2.3.

We see from changing S values that introducing even a small amount of heterogeneity has meant that the estimate of S is poor and the confidence intervals don't contain the true value in any case. We also see that the P estimates are very poor and have much wider confidence intervals than we have seen with homogeneous data. When changing P, we see that the S estimates are reasonable, but the P estimates are again very poor, with the true values not appearing in the confidence intervals.

It is thus our suggestion that if significant heterogeneity in the data is present, as may be the case for the experimental data here, then a model different from the PRW model should be used in the framework to ensure that estimates are not biased in this way. It is worth noting though that the estimates from the experimental data are not as poor as the ones studied in these scenarios, particularly where S is concerned.

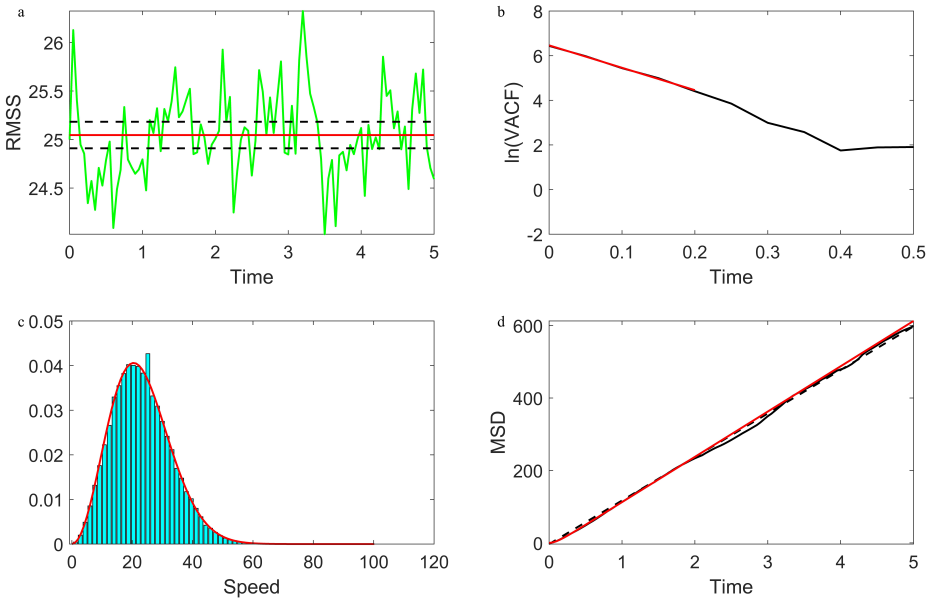

Figure 2.6: **Testing 3D *in silico* output based on experimental parameters** for 550 cells over 100 simulation periods with $dt = 0.05$ and $S = 25, P = 0.1$. Cells are initialised at the origin, $\mathbf{x}_0 = \mathbf{0}$, with speed S and orientation sampled uniformly from the unit sphere. **a)** $RMSS$ over time is shown (green line) with estimated average speed $\hat{S} = 25.0458$ (red line) and 95% confidence interval $[24.9091, 25.1825]$ ($n_e = 40$) (black dashed lines). **b)** Calculated $\ln(VACF)$ vs time (black line) with FGLS line fit (red line) giving $\hat{P} = 0.0996$ with 95% confidence interval [0.0978, 0.1015]. **c)** Histogram of speeds with Maxwell-Boltzmann density with parameter S overlaid (red curve). **d)** Calculated MSD vs time (black line) with model predicted MSD (red line) and a straight line fitted to the calculated MSD (black dashed line). The inferred P estimate from the MSD calculations is $\hat{P} = 0.0962$.

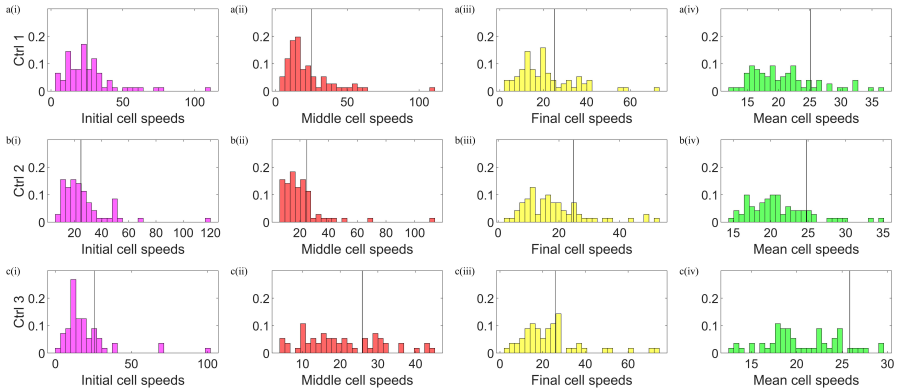

Figure 2.7: **Speed distribution for 3D cell tracks lasting the full length of the experiment,** for **a)** Spheroid 1, **b)** Spheroid 2, **c)** Spheroid 3. **i** initial speed distribution, **ii** intermediate speed distribution, **iii** final speed distribution, **iv** mean cell speed distribution. Vertical black lines display the estimate for $2S\sqrt{\frac{2}{3\pi}}$, the mean value of the theoretical Maxwell-Boltzmann distribution. Each row corresponds to an independent spheroid.

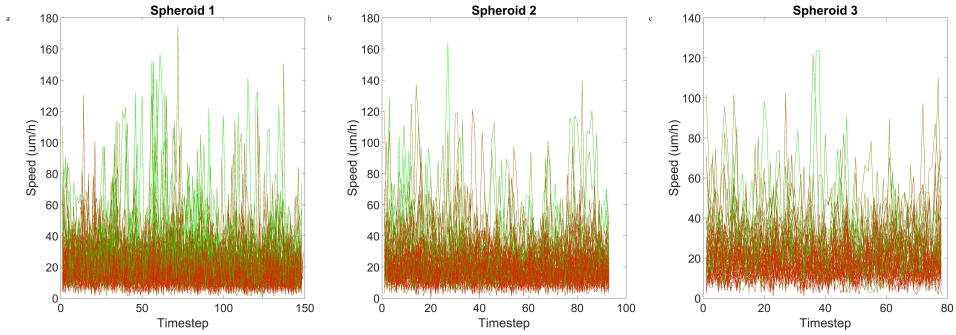

Figure 2.8: **Plots of individual cell speeds at each 3 minute time step** for **a)** Spheroid 1, **b)** Spheroid 2, **c)** Spheroid 3. Speed data taken from Richards *et al.* (2018). Speeds are plotted for individual cells whose velocities were recorded for the full duration of each experiment. Cells are colour coded to represent their mean speed, with green lines representing cells with higher mean speeds, red lines cells with lower mean speeds.

2.2 Using the PRW model to describe cell motility in 2 dimensions

2.2.1 *In silico* tests

The main focus of the modelling work has been in 3D due to the novelty it provides in studying 3-dimensional cell motility in a rigorous and data-driven manner, using the framework.

In this section, the 2-dimensional version of the framework is presented for completeness and to demonstrate how one could use the framework in 2D if required, though the approach is the same. Necessary adjustments are made to the PRW model formulation and statistical measures due to the change in dimension in that which follows and as such the important formulae are restated here.

The code which runs the framework in 2D along with all functions is available online via the link given in Appendix A.

For 2-dimensional modelling the PRW SDE used is as in equation 1.5 with the dimensional diffusion coefficient $D_2 = S^2P/2$, giving

$$d\mathbf{v} = -\frac{1}{P}\mathbf{v}\,dt + \frac{S}{\sqrt{P}}\,d\mathbf{W}(t). \tag{2.11}$$

The 2-dimensional analogues of the statistical measures are also adjusted. The S estimates will be calculated based on the $RMSS$ as

$$\hat{S} = \frac{1}{T}\sum_{t=0}^{T} RMSS(t) = \frac{1}{T}\sum_{t=0}^{T}\sqrt{\langle v_x(t)^2 + v_y(t)^2\rangle},$$

with confidence intervals given as for the 3-dimensional case, using the expressions in equations 2.4, 2.5 and 2.6.

The theoretical MSD is unaffected by dimension and so is still given by

$$MSD(t) = 2S^2P^2\left(e^{-\frac{t}{P}} + \frac{t}{P} - 1\right),$$

though the formula for direct calculation of the quantity from the data is now given by

$$MSD(t) = \left\langle (x(t) - x(0))^2 + (y(t) - y(0))^2 \right\rangle,$$

with $(x(t), y(t))$ being the position vector of the cell at time t and $\langle . \rangle$ being the average over all cells.

Estimates of P will again be inferred from the MSD using the slope of the FGLS regression line fitted to the calculated MSD vs t plot.

The theoretical 2-dimensional velocity autocorrelation is also unaffected by the dimensional change and is given by

$$VACF(t) = S^2 e^{-\frac{t}{P}}$$

and the formula for calculating this correlation function from the data is

$$VACF(t) = \langle (v_x(0) \cdot v_x(t)) + (v_y(0) \cdot v_y(t)) \rangle ,$$

again with $\langle . \rangle$ being the average over all cells, and $(v_x(t), v_y(t))$ being a cell's velocity vector at time t. As in the 3-dimensional case confidence intervals will be calculated for estimates of P using equations 2.9 and 2.8.

Finally the speed distribution cells are expected to follow in 2 dimensions is the Rayleigh distribution with scale parameter $\sigma = S/\sqrt{2}$ and speed u, given by

$$f(u; S) = \frac{2u}{S^2} e^{-\frac{u^2}{S^2}}.$$

2.2.2 Output from *in silico* simulations

As with the 3D framework, some examples of output from *in silico* simulations are shown here for 2D cell tracks. Figure 2.9 shows an example 2 dimensional data set with $S = 1$, $P = 1$, and figure 2.10 shows an example where $S = 25$, $P = 2$. Both figures show good fit of the model to the data, and estimates from the framework are given in table 2.1, along with the inferred P and S estimates from the MSD and $\ln(VACF)$ regression models, respectively. The framework estimates are accurate, though the true values of some parameters seem to be right on the edge of confidence intervals. The MSD inferred P estimates are less accurate than the estimates provided by the $\ln(VACF)$ estimates.

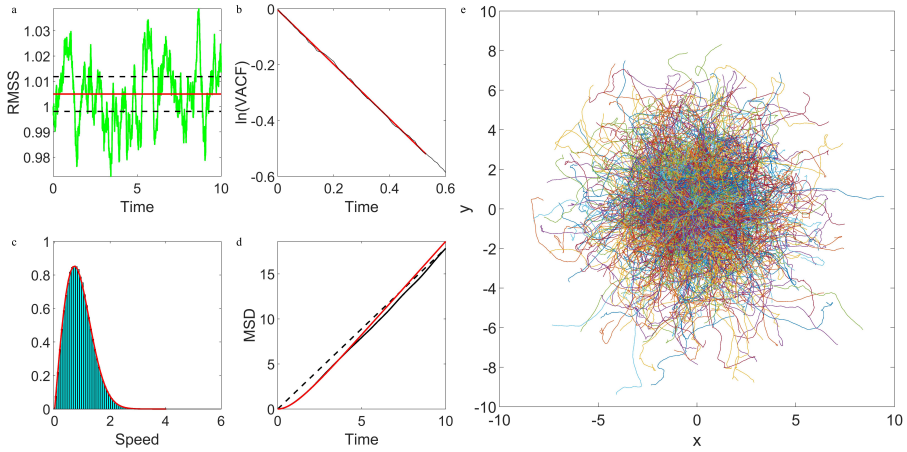

Figure 2.9: **Simulation output for 2D cell tracks** for 1000 cells over 1000 simulation periods of $dt = 0.01$ and with $S = 1, P = 1$. Cells are initialised at the origin, with speed S and orientation sampled uniformly from the unit sphere. **a)** $RMSS$ over time is shown (green line) with estimated average speed $\hat{S} = 1.0050$ (red line) and 95% confidence interval [0.9981, 1.0118] (black dashed lines). **b)** Calculated $\ln(VACF)$ vs time (black line) with FGLS line fit (red line) giving $\hat{P} = 1.0261$ with 95% confidence interval [1.0088, 1.0441]. **c)** Histogram of speeds with Maxwell-Boltzmann density with parameter S overlaid (red curve). **d)** Calculated MSD vs time (black line) with model predicted MSD (red line) and a straight line fitted to the calculated MSD (black dashed line). The inferred P estimate from the MSD calculations is $\hat{P} = 0.8827$.

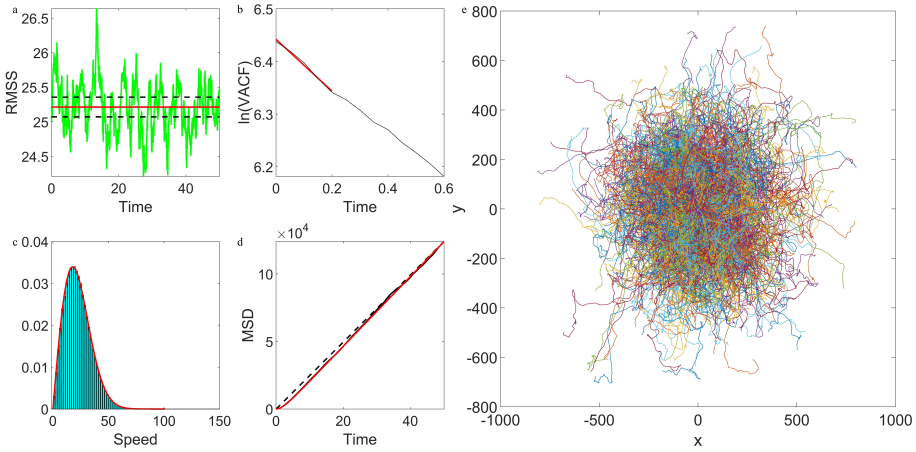

Figure 2.10: **Simulation output for 2D cell tracks** for 1000 cells over 1000 simulation periods of $dt = 0.05$ and with $S = 25, P = 2$. Cells are initialised at the origin, $\mathbf{x}_0 = \mathbf{0}$, with speed S and orientation sampled uniformly from the unit sphere. **a)** $RMSS$ over time is shown (green line) with estimated average speed $\hat{S} = 25.2132$ (red line) and 95% confidence interval [25.0702, 25.3562] (black dashed lines). **b)** Calculated $\ln(VACF)$ vs time (black line) with FGLS line fit (red line) giving $\hat{P} = 2.0327$ with 95% confidence interval [1.7412, 2.4416]. **c)** Histogram of speeds with Maxwell-Boltzmann density with parameter S overlaid (red curve). **d)** Calculated MSD vs time (black line) with model predicted MSD (red line) and a straight line fitted to the calculated MSD (black dashed line). The inferred P estimate from the MSD calculations is $\hat{P} = 1.9518$.

2.2.3 Applying the framework to 2-dimensional experimental tracking data

Specifics of the experimental data

Experimental data was collected by Light Sheet Fluorescence microscopy of U87 glioblastoma tumour cells in a control environment. Images were again taken every 3 minutes, this time over a 48 hour period, meaning that there are 960 periods of 0.05 hours in the data set.

Data were formatted so that all trajectories were used, no matter when in the experiment they began, thus T is time over a trajectory instead of 'true' time over the course of the experiment. All quantities were converted to reflect units of μm/h, the Ncells parameter was decided based on the number of unique track IDs in the data set, and velocities are calculated from positions using $(x(t+1) - x(t))/dt$. Code for formatting this data can be found online with details given in Appendix A.

Parameter estimation and goodness-of-fit for the experimental data

The experimental data was run through the framework, however estimates obtained were extremely poor. To demonstrate this an example of the plots and estimates is provided for one of the experimental replicates. For this data set the framework gave estimates of $\hat{S} = 2.5928\,\mu$m/h, $[0.5426, 4.6430]$ and $\hat{P} = -17.6078\,$h, $[-54.8113, -10.4886]$, as detailed in table 2.2. We are confident that the estimates of speed are accurate given that this a quantity that can be measured easily and validated by experimentalists. It is however concerning that the confidence interval for $\hat{S}$ is so wide, and we notice again, as in the 3D case, that the $RMSS$ seems to increase over time, likely affecting this estimate of S which relies on the $RMSS$ fluctuating around some stationary mean value.

The estimate for $\hat{P}$ is even more concerning. The negative values come from the fact that the slope of the $\ln(VACF)$ graph is positive in this example, as seen in figure 2.12b). Clearly this value of P is not possible, as negative persistence time does not make sense.

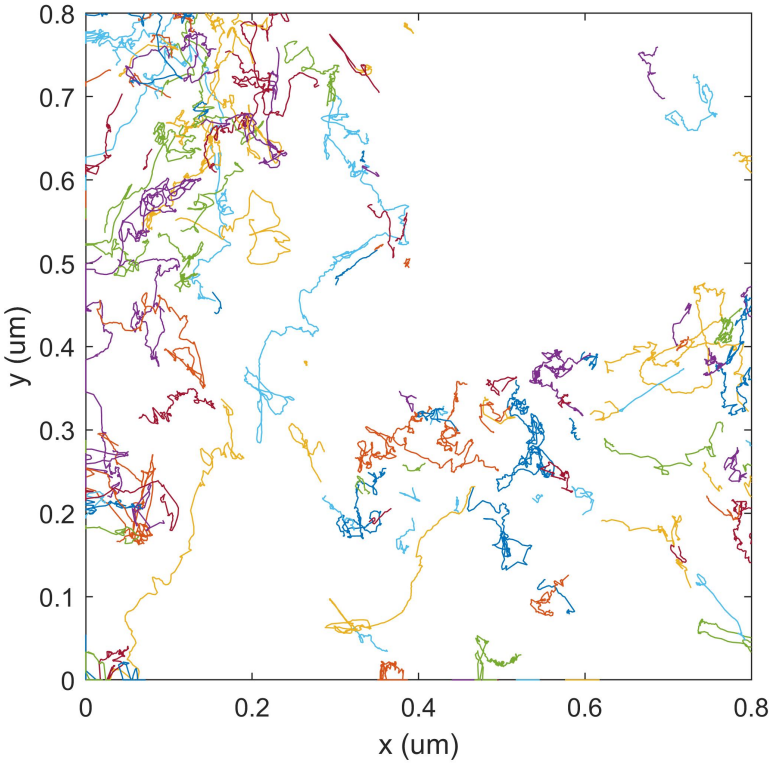

Figure 2.11: **Example plot of 2D experimental cell tracks** for 143 cells over 960 simulation periods of $dt = 0.05\,\mathrm{h}$.

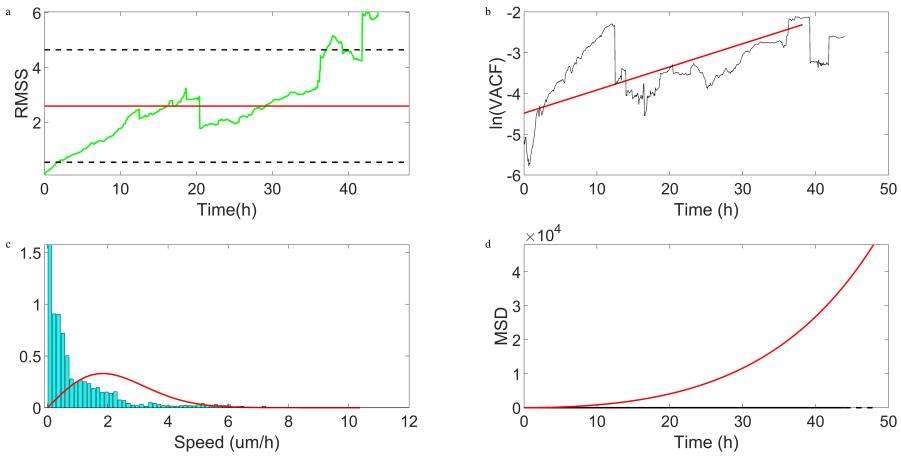

Figure 2.12: **Framework output for 2D experimental cell tracks** for 143 cells over 960 simulation periods of $dt = 0.05\,\text{h}$. **a)** $RMSS$ over time is shown (green line) with estimated average speed $\hat{S} = 2.5928\,\mu\text{m/h}$ (red line) and 95% confidence interval [0.5426, 4.6430] (black dashed lines). **b)** Calculated $\ln(VACF)$ vs time (black line) with FGLS line fit (red line) giving $\hat{P} = -17.6078\,\text{h}$ with 95% confidence interval [-54.8113, -10.4886]. **c)** Histogram of speeds with Maxwell-Boltzmann density with parameter S overlaid (red curve). **d)** Calculated MSD vs time (black line) with model predicted MSD (red line) and a straight line fitted to the calculated MSD (black dashed line). The inferred P estimate from the MSD calculations is $\hat{P} = 0.00024\,\text{h}$.

The rest of the plots in figure 2.12 also show how poor the model fits to the data, with the experimental speed distribution in figure 2.12c) having a peak much further to the left than what the model predicts and the model MSD being a vast overestimate of the calculated MSD, seen in figure 2.12d).

To probe this further, we inferred $\hat{P}$ from the MSD and got $\hat{P} = 0.00024\,\text{h}$. This value is much smaller than $dt = 0.05\,\text{h}$ and so this makes it clear that the experimental data set violates the assumption that $dt << P$, needed to observe correlation over enough time steps to observe the decay. This means that any correlation in the velocity is disappearing within any given time step. We see from equation 2.11 that if the correlation time is very small, the acceleration will be very large and thus velocity will change extremely quickly and no persistence will be seen.

Upon generating *in silico* data with parameter values $S = 2$ and $P = 0.0002$, it was found that the framework could not calculate estimates for P. This was because the cut-off algorithm on the $\ln(VACF)$ vs t plot could not run due to there being less than 5 data points that met the criteria for successful cutting of the data. This is most likely due to the fact that P here is so small compared to dt that any correlation in the velocity disappears within any given time step and is subsequently not observed in the $VACF$.

It thus appears that the sampling rate in this experimental data is insufficient for studying this data with our framework, though it is through the rigour used that we are able to diagnose the problem as the correlation time being too short to be observed. We thus suggest two things for analysing cell tracking data such as this. Firstly that where possible experimentalists sample frequently enough that correlation in velocity can be seen in the $VACF$, and secondly that this may be helped by using the $\hat{P}$ inferred from the easily calculable MSD and $\hat{S}$. This way we can gain an understanding of the magnitude of $\hat{P}$ and check that it is appropriate and satisfies assumptions of the PRW model before trying to use the framework. This allows experimentalists and mathematical modellers to work in conjunction to come up with reliable and accurate estimates from the framework, having first checked assumptions, and test motility hypotheses as

desired.

2.3 Discussion and Conclusions

Chapters 1 and 2 have presented an example of a rigorous combined mathematical and statistical approach for analysis of 2- and 3-dimensional cell tracking data using stochastic models. The framework we have developed provides tools for calculating various statistical measures for testing goodness-of-fit and for parametrising the given model, here demonstrated using the Persistent Random Walk model. This model has been chosen in the knowledge that it is perhaps not complex enough to fully capture the motility seen in GBM spheroids, but is one of the most popular stochastic models used in cell motility.

The ill-fitting nature of the model though allows us to exploit the framework and show its potential in uncovering features of a data set that may be missed by less rigorous analysis or a more well-fitting, but not optimal model. We also make clear the distinction between the PRW model in all 3 physical dimensions, and how the governing equation changes based on the dimension-dependent diffusion coefficient, something which has not previously been stated as clearly.

Our framework outputs parameter estimates along with confidence intervals and uses statistical measures to provide them, all of which take into account serial correlation in the data. This has been lacking in the literature in this context until now, to the best of our knowledge. We believe the approach we present is adaptable to other models and data sets by simulating *in silico* data sets using the model of choice and using statistical measures appropriate for the data being studied in the same way we have demonstrated to compare the model with the data. It is the consistency and thoroughness of the approach which allows for elucidation of possible reasons for mismatch between a model and a data set, but also suggest routes for further exploration of 3D cell tracking data sets such as the ones explored here.

The framework as a package is useful for experimentalists looking to analyse tracking data without necessarily having the mathematical or statistical background required to carry out such rigorous analysis. There is also great benefit

for modellers in being able to test potential models for a data set with the same consistent, thorough method of testing, allowing for direct comparison of population level statistics between models. There are also benefits to those looking at initial analysis of a data set before moving on to more complex considerations by way of the plots that the framework outputs, as well as quantitative descriptions of population and individual track characteristics such as speeds and correlations in velocity.

The framework has been tested on *in silico* data sets in 2 and 3 dimensions through the use of statistical measures MSD, $VACF$ and speed histograms, before being applied to experimental cell tracking data collected from GBM tumour spheroids. Results show that the PRW model may not be complex enough to describe these particular data sets well, as shown by the experimental data having different speed distributions and MSDs than predicted by the model in the 3D case.

Though others have reached this conclusion before for other cells types (Wu *et al.*, 2014; Dieterich *et al.*, 2008; Metzner *et al.*, 2015; Upadhyaya *et al.*, 2001; Cherstvy *et al.*, 2018; Loosley *et al.*, 2015), we have done so with statistically significant proof and through following the same rigorous procedure for each data set. These findings allow us to question what about the model itself and the biology needs further investigation, likely taking into account the proliferative and heterogeneous nature of cancer cells. For example our investigation of the cell speed distribution allows us to see that although the bulk of the cells are travelling as we expect, there are some outliers moving particularly fast and there is surely an interesting biological reason behind this.

In the 2D case the magnitude of the persistence time proved to be too small in the experimental data sets to allow us to run data through the framework. Through the rigour of this framework we were able to diagnose this as being a problem for further analysis using the PRW model, and still obtain an estimate for persistence time using the MSD. We suggest that experimentalists use the MSD method to estimate persistence time before trying to use the framework as a whole to ensure that assumptions of the PRW model are met, particularly

that cells persist over enough time steps for correlation in velocity to be detected by the $VACF$.

In these analyses, we were grateful to have large data sets to work with, though we excluded any 3D tracks that did not begin at the start of the experiment, greatly reducing our available data. We were however still able to reach the conclusions above and reject the PRW model for all experimental data sets, with strong evidence to back this up. This demonstrates that it is not the amount of data that is vital here, but experimental parameters such as the frequency with which measurements are taken, and the length over which cells are studied. This is just one example of how iterating between mathematical models and experiments will elucidate new directions for modelling and study of biological systems.

Cancer is a complex condition in which cells interact with each other and with many other molecules within a tumour microenvironment. Cells can also vary between themselves, and are capable of changing their own behaviour in response to certain stimuli. This presents a problem with creating models simple enough to test certain motility hypotheses for a data set such as the one we have been working with, given the wide range of conditions that would need to be taken into account. This challenge only increases when drugs are brought into the system and so there is plenty of scope for the model to be adapted to incorporate any or a range of these complications. We do however see the potential of a framework such as ours to be able to estimate motility parameters under different conditions, for example, when spheroids are treated with drugs.

In future work we would endeavour to consider problems outlined throughout this chapter such as ensuring the suitability of experimental data for this framework and how to make use of all available data when carrying out the analysis. We could further consider adapting the model so that the alternative ideas about MSD, $VACF$ and velocity distributions may be studied rigorously (Yurchenko *et al.*, 2019; Luzhansky *et al.*, 2018; Takagi *et al.*, 2008). We would also hope to be able to add alternative terms into the model to better describe how the cells are moving in response to chemical stimuli in addition to random motion.

Cells are known to have a 3-step migration cycle (Lauffenburger & Horwitz, 1996; Mitchison & Cramer, 1996) consisting of protrusion of the cell's leading edge, adhesion of this region to the underlying substrate and then contraction of the cell body causing detachment of the rear of the cell, which has been incorporated into some cell migration models, though cells in 3D change the mode of their migration depending on the geometry of their environment (Wolf *et al.*, 2013; Wu *et al.*, 2018; Mierke, 2015).

A lot of the differences between 2 and 3D cell migration are as a result of the Extracellular Matrix (ECM) which surrounds cells in 3D. For example, the availability of space for cells to move through (Wolf *et al.*, 2013; Tozluoğlu *et al.*, 2013), resistance cells face from the ECM, the viscosity and stiffness of the matrix (Zaman *et al.*, 2005, 2006; Wang *et al.*, 2014), and the presence or absence of matrix proteins (Fraley *et al.*, 2015; Wu *et al.*, 2018) can all affect the migration of a cell in 3D. Thus, following suit of others in the field, incorporating terms that describe the influence of the ECM would no doubt improve the model fit, as well as considering other phenomena such as chemotaxis, haptotaxis and gradients in nutrients, and more specific to cancer, angiogenesis, hypoxia and necrosis.

It would also be informative to add cell-cell interactions into the model, as this may be one of the reasons for the mismatch between the PRW model and the experimental data. In order to study this further one could look to models of cell motility that include interaction terms such as the Vicsek model (Vicsek *et al.*, 1995; Cziròk *et al.*, 1999; Liu, 2010), that of Sepúlveda *et al.* (2013) or of Matsiaka *et al.* (2019). Generating *in silico* data from any of these models and running this data through the framework would reveal whether interactions do need to be included in the model, evidenced by further mismatch. This would allow rigorous study of how interactions affect the statistical measures and parameter estimates and potentially suggest sensible avenues of exploration for alternative models to place within the framework.

For now, we present this framework as a data-driven, rigorous methodology for testing whether a cell tracking data set could reasonably be described by

a given model. It provides statistical measures for assessing how realistic the model is for a data set, and tests whether we can obtain estimates of population level parameters using individual cell properties, paving the way for future interrogation of cell tracking data and investigation of cell motility, particularly in 3D where there is a lack of tools for such rigorous analysis.

Chapter 3

A Bayesian approach to estimating cell motility parameters using the Persistent Random Walk model

3.1 Introduction

3.1.1 Bayesian Ideas

Bayesian methodology can be seen as an alternative way of thinking from the more widely adopted frequentist or classical statistical mindset. Conceptually, the frequentist approach sees unknown parameters as random and uncertain, but as fixed quantities. It then looks at finding the best model for a set of observed data by picking an optimal parameter set that describes this data. The result is a set of point estimates of parameter values, usually with some surrounding confidence interval. Probability in this school of thought is thought of in terms of the frequency of a repeated event occurring within a large number of trials.

The Bayesian approach instead views probability as a description of uncertainty and so doesn't require large sample sizes or repeated events to have con-

fidence in analytical methods. If there is already observed data, the Bayesian school of thought sees this as fixed and seeks to fit a model to the data with model parameter values chosen according to what is judged the most likely given the observed data. Unknown quantities are viewed as random variables with their own distribution functions.

Bayesian analysis takes information from both prior distributions and likelihood functions for the observed data and gives a posterior distribution for the parameter values being estimated, naturally taking into account uncertainty in the estimated values. Various quantities can then be extracted from this posterior distribution, including marginal densities for each parameter, which can be used in subsequent inference from the fitted model. The way that parameters are treated in Bayesian analyses lends itself to natural descriptions of things like p-values and credibility intervals, concepts which are often wrongly described from a frequentist point of view.

The arguments used in Bayesian analysis were first introduced by Thomas (Bayes, 1763) in an essay to the Royal Society, and by (Laplace, 1812) in his book 'Analytic Theory of Probability'. Bayesian methods predate frequentist ones and the objective ideas of Bayes, Laplace and others about priors remained in use for 200 years before others started to develop the field further. The modern development of Bayesian methods started in the 1950s and 1960s (O'Hagan, 2004) and has continued more rapidly since the 1990s (Ashby, 2006).

Central to Bayesian analysis is Bayes' Theorem (Bayes, 1763) which says that the probability of an event A given an event B has happened is given by

$$P(A|B) = \frac{P(A \cap B)}{P(B)}, \tag{3.1}$$

for $P(B)$ the probability of B happening, $P(A|B)$ the probability of A given B, and $P(A \cap B)$ the probability of A and B intersecting.

This could also be written as

$$P(B|A) = \frac{P(A \cap B)}{P(A)} \implies P(A \cap B) = P(B|A)P(A). \tag{3.2}$$

So rewriting equation 3.1 using equation 3.2 gives

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}. \tag{3.3}$$

When looking to estimate parameters one can first look at how Bayes' theorem uses prior information along with likelihood to provide a posterior density. For a data set x and a model M with parameters $\Theta = (\theta_1, ..., \theta_n)$ which is to be fitted, we look at the prior $P(\theta_i)$ for parameter θ_i, a probability density representing prior beliefs about the value of θ_i. It is desirable to know the posterior distribution for each parameter, $P(\theta_i|x)$, a description of the probability of parameter θ_i given the data. The likelihood function for each parameter, $P(x|\theta_i)$, is also of interest and is defined as the probability of observing the data we have, given the parameter value θ_i.

Putting all of this together using equation 3.3 yields

$$P(\theta_i|x) = \frac{P(x|\theta_i)P(\theta_i)}{P(x)},$$

where the quantity $P(x)$ is called the evidence or the marginal likelihood, the probability of the data being generated by the proposed model.

Thus to obtain a posterior density for a model, priors are combined with likelihood, updating current beliefs and giving a complete picture of uncertainty in any resulting parameter estimates.

The main goal of a Bayesian analysis is to ensure that the outputs are conditional on the data that has been obtained whilst respecting the frequentist notion that the methodology must ensure success upon repetition of the analysis (Berger, 2006). A brief outline of the process by which a Bayesian analysis takes place can be thought of as (O'Hagan, 2004):

- create a statistical model to link data to parameters and give the likelihood function
- formulate prior information about the parameters
- combine these two sources of information using Bayes' Theorem
- use the resulting posterior to derive inferences about parameters

The use of likelihood functions is a commonality between frequentist and Bayesian methods, with the frequentist approach often employing only Maximum Likelihood Estimation, but with Bayesian methods combining this with

information from the data in the form of priors. The choice of priors in a Bayesian analysis is therefore no doubt one of the most important and influential parts of the analysis. An ideal situation is one in which there is a wealth of information available about parameters of interest, allowing informative priors to be constructed and used in subsequent analysis. This can be achieved by consulting with subject matter experts and elucidating the best choice of prior distribution based on this expert knowledge.

However, this information is not always available and so uninformative or diffuse priors are often used to represent a lack of information surrounding parameter values, whether this is due to random variation in the parameters themselves or just imperfect knowledge of their values. For analyses involving several parameters to be estimated it is often expected that some parameters will be given informative priors, but that others will be given uninformative priors as there is a lack of detailed knowledge about plausible or realistic values for them.

A commonly used uninformative and objective prior is Jeffreys' prior (Jeffreys, 1946) which suggests the prior density be proportional to the square root of the determinant of the Fisher information matrix for the parameters. This matrix tells us about the amount of information that a random variable Y contains about the unknown parameters θ. The related notion of reference priors was introduced by Bernardo (1979) and further developed in Berger & Bernardo (1992), aiming to objectify the choice of priors by providing standard priors for a variety of specific contexts and statistical models.

The main 'result' of the Bayesian analyses will be a joint posterior density for parameters to be estimated. This gives an overall picture of the uncertainty surrounding parameter values, and allows estimation and inference to be carried out directly on parameters of interest when marginal posteriors are used. The resulting marginal posterior densities for each parameter can provide credibility intervals which allow the calculation of an interval with the probability of the true parameter value being in the given interval equalling the significance level of this interval.

This is the interpretation incorrectly given often to confidence intervals ob-

tained with frequentist analysis. But, a 95% confidence interval, for example, suggests that if the experiment was repeated many times and a 95% confidence interval for the parameter of interest calculated each time, then about 95% of such intervals would contain the true parameter value.

Upon obtaining the posterior density, we could also look at the posterior predictive density which gives the distribution of possible unobserved values, conditional on those observed. This is done by using the posterior distribution of the parameters, and the likelihood for the unobserved value and then marginalizing the density for this unobserved value over the posterior density.

3.1.2 Bayesian Methods: Thomas Bayes to the present day

As mentioned previously, the Bayesian school of thought was introduced by Thomas Bayes when his essay on the topic was published posthumously in Bayes (1763). This work was furthered by Laplace (1812) and largely remained in this state for over 100 years. In the 1950s the field picked up interest, with development and refinement of the methods and techniques greatly increasing thereafter (Ashby, 2006).

The biggest growth in application and study of Bayesian methodology however has been since the 1990s with the advent of increased computational power allowing more sophisticated methodologies to be developed for solving problems the 'Bayesian way'. Previously, the problem was that in cases when posterior distributions could not be written down analytically, usually in multi-dimensional problems, analysis of the posterior was often impossible.

In practice, likelihood functions can also be difficult to find, especially in cases where the distributions of the priors and posteriors are not conjugate. The prior is called conjugate with the likelihood function if the combination of these two elements produce a posterior distribution within the same family as the prior. For a conjugate prior, the form of the posterior is already known and thus parameters of this distribution can be found easily.

The advent of new computational methods allowed the study of the posterior

even in cases where it was unknown or could not be written analytically. This has since opened up new avenues of exploration and has seen new ways to get around issues that had been causing Bayesians problems for decades. These computational methods can broadly be split into those based on Markov Chain Monte Carlo, or MCMC, and those not using MCMC. An overview of current methodologies for solving problems 'the Bayesian way' is provided below, with the illustration of some selected methods that are used most widely.

MCMC-based methods

The seminal work by Gelfand & Smith (1990) introduces the idea of using Markov Chain Monte Carlo (MCMC) simulations as a way to take large samples from a target distribution, the stationary distribution of the Markov chain.

Monte Carlo methods are ways of using simulation to approximate variables or quantities of interest when they are difficult or impossible to obtain analytically. At the heart of these methods is repeated sampling from probability distributions that are assumed of the uncertain parameters in the proposed model.

A Markov chain is formed from a series of states that create a random walk through a parameter space. The chain moves according to its transition kernel, a matrix of probabilities that determine the probabilities of transitioning between states, and most importantly the current state depends only on the one before it. This chain converges to its stationary distribution, where it will remain, moving around within this distribution.

The idea of MCMC is to combine Monte Carlo simulation and Markov chains to get around the difficulty of finding the marginal likelihood by constructing a Markov chain with the distribution of interest as the stationary distribution. This is done by repeated sampling as in Monte Carlo simulation, ultimately exploring the stationary distribution of the chain which should be the desired posterior. This technique allows sampling from the posterior distribution without actually knowing what it is, essentially having as good as the whole posterior density from which to directly calculate statistics of interest.

The Markov chain created should be a continuous discrete-time Markov chain

since it will run over the parameter space and sample continuous parameter distributions at discrete time steps. After creating the chain with a transition kernel such that the stationary distribution is the target posterior, an acceptance ratio is calculated as the chain moves between possible parameter values. This circumvents the need to calculate the complicated marginal likelihood $P(x)$ by comparing the posterior probabilities for the current and proposed parameter values at each step. Namely, for current parameter value θ_0 and proposed parameter value θ, the acceptance ratio is calculated as

$$\frac{P(\theta|x)}{P(\theta_0|x)} = \frac{\frac{P(x|\theta)P(\theta)}{P(x)}}{\frac{P(x|\theta_0)P(\theta_0)}{P(x)}} = \frac{P(x|\theta)P(\theta)}{P(x|\theta_0)P(\theta_0)}.$$

As becomes clear, the marginal likelihoods cancel out and what is left are probabilities that are easy enough for a computer to calculate given the data set and prior distributions.

The acceptance ratio is used at each state of the Markov chain, and allows us to answer the question "Does the proposed parameter value explain the data better than the current value?". The acceptance ratio helps answer this question by quantifying the relationship between the two parameter values. If the ratio is greater than 1, definitely go to the proposed value. If the ratio is less than one, then we make the jump to the proposed value with a non-zero probability. Generally if the ratio is less than 1 then the proposed parameter value is visited rarely or less often whereas regions with higher probability are visited relatively more often.

It is common when conducting MCMC simulations to allow a period of 'burn-in' which involves discarding the first n samples in the chain taken during this burn-in period and only calculating subsequent statistics on the rest of the samples taken. This allows the chains to mix well and sample the whole of the parameter space of interest.

Metropolis-Hastings Algorithm MCMC-based methods are largely based on the Metropolis-Hastings (M-H) algorithm, with all of the methods outlined below being special cases of this seminal algorithm. The M-H algorithm was first proposed by Metropolis *et al.* (1953) and was later generalized by Hast-

ings (1970). Being based on a Markov chain, it works by generating a sequence of samples in such a way that the distribution of these sampled values converges to the distribution of interest - the posterior. It uses the current parameter values to choose the next ones based on an acceptance probability.

The algorithm will always accept values that increase the posterior probability, and will accept values that decrease the posterior probability with some non-zero probability. If the candidate value is accepted, the next step in the chain uses this candidate value and, if it is rejected, then the current value is used again in the next step.

The M-H algorithm can draw samples from any probability distribution $P(\mathbf{x})$, for parameters $\mathbf{x} = x_1, ..., x_N$, provided that there is a known function $f(\mathbf{x})$ proportional to the density of $P(\mathbf{x})$. The values of this function can be calculated, making it useful when directly calculating the posterior distribution is difficult.

To carry out the algorithm, it is necessary to start with a proposal candidate-generating density, from which candidate values are generated. According to Metropolis *et al.* (1953), this proposal distribution should be symmetric, though Hastings (1970) generalised the algorithm to non-symmetric proposals, increasing the convergence speed. The proposal distribution should however be easy to sample from, ensuring that jumps in the algorithm can cover the parameter space in a reasonable time and jumps are not rejected too frequently.

It is common to consider proposal distributions close to the target distribution for this reason, and to use normal jumps centred around the previous chain position. The variance of this proposal density is thought of as a tuning parameter and can be adjusted accordingly to improve mixing and convergence of the chain (Walsh, 2002). A small variance would see a high acceptance rate but slow mixing due to successive values being very close together, but if the variance is too high the chain will likely visit areas of low density more often thus giving a low acceptance rate, still with slow mixing.

Briefly, the M-H algorithm is as follows:

- Start with initial state $\mathbf{x_0}$

For $i = 1, 2, ..., k$, where k is the length of the Markov chain:

- Generate a candidate state $\mathbf{x_i'}$ according to proposal density $p(\mathbf{x_i'}|\mathbf{x_i})$, for current state $\mathbf{x_i}$
- Calculate the acceptance probability of a move from state $\mathbf{x_i}$ to $\mathbf{x_i'}$

$$\alpha(\mathbf{x_i}, \mathbf{x_i'}) = \min\left(1, \frac{\pi(\mathbf{x_i'})}{\pi(\mathbf{x_i})}\frac{p(\mathbf{x_i}|\mathbf{x_i'})}{p(\mathbf{x_i'}|\mathbf{x_i})}\right),$$

  for stationary (target) distribution $\pi(.)$
- Generate u_i from Unif$(0, 1)$
- Accept the move from state $\mathbf{x_i}$ to $\mathbf{x_i'}$ if $u_i \leq \alpha(\mathbf{x_i}, \mathbf{x_i'})$, otherwise reuse the current state $\mathbf{x_i}$
- Return $\{\mathbf{x_k^{(1)}}, ..., \mathbf{x_k^{(N)}}\}$ as the sample from the target density $\pi(\mathbf{x})$

Gibbs Sampler The Gibbs sampler is one of the special cases of the M-H algorithm, and was first introduced by Geman & Geman (1984). It forms the basis of MCMC sampling as introduced by Gelfand & Smith (1990) and has been widely used since in Bayesian analysis, including in popular software packages for MCMC sampling such as BUGS (Gilks *et al.*, 1994), JAGS (Plummer, 2003) and NIMBLE (de Valpine *et al.*, 2017), all of which are based on the BUGS language.

The Gibbs sampler differs from the M-H algorithm in that it considers the distribution of each parameter of interest conditional on the other parameters of interest. The sampler is of use when it is difficult to sample from the desired distribution, but sampling from conditional distributions is possible, for example, for two parameters of interest x and y, $P(x, y)$ is difficult to sample from but sampling from $P(x|y)$ and $P(y|x)$ is easier.

A simple example of where the Gibbs sampler may be advantageous is given by Huerta (2012). Suppose we have unknown parameters n and θ where the prior for n is given by $g(n) \sim$ Poisson(λ) for some known λ and the prior for θ is given by $g(\theta) \sim$ Beta(a, b) for some known a and b. Also suppose there is a binomially

distributed random variable $X \sim \text{Bin}(n, \theta)$ for which we want to estimate the marginal distribution $P(x)$. The joint distribution of X, θ, n is given by

$$P(x, \theta, n) = \binom{n}{x} \theta^x (1-\theta)^{n-x} \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} \theta^{a-1} (1-\theta)^{b-1} \left(e^{-\lambda} \frac{\lambda^n}{n!} \right),$$

for $x = 0, ..., n$, $0 < \theta < 1$ and $n = 0, 1, 2, ...$ The marginal distribution of X is impossible to find analytically here, but the conditional densities are easily written down analytically. Disregarding constants, the above density can be written as

$$P(x, \theta, n) \propto \binom{n}{x} \theta^{a+x-1} (1-\theta)^{b+n-x-1} \frac{\lambda^n}{n!}.$$

We can then write the full conditionals needed for the Gibbs sampler as

$$\begin{aligned}
P(x|\theta, n) &\propto \binom{n}{x} \theta^x (1-\theta)^{n-x} \propto \text{Bin}(n, \theta), \\
g(\theta|x, n) &\propto \theta^{a+x-1} (1-\theta)^{b+n-x-1} \propto \text{Beta}(a+x, b+n-x), \\
g(n|\theta, x) &\propto \binom{n}{x} \frac{\lambda^n}{n!} (1-\theta)^{n-x}.
\end{aligned}$$

Samples of θ and n would then be taken in turn from the respective conditional distributions at each iteration of the algorithm.

In the more general case for N parameters, if the conditionals can be written as above, to carry out the Gibbs sampling algorithm we would set initial values $x_1^0, ..., x_N^0$ for the parameters, $x_1, ..., x_N$, then proceed to sample from the conditional distributions for iteration i from 1 to k as follows:

Sample x_1^i from $P(x_1^i | \, x_2^{i-1}, ..., x_N^{i-1})$

Sample x_2^i from $P(x_2^i | \, x_1^i, x_3^{i-1}, ..., x_N^{i-1})$

.

.

.

Sample x_N^i from $P(x_N^i | \, x_1^i, x_2^i, ..., x_{N-1}^i)$

where, for example, $P(x_j^i | \, x_1^i, ..., x_{j-1}^i, x_{j+1}^{i-1}, ..., x_N^{i-1})$ is the distribution of x_j^i, for some $1 < j < N$ conditioned on all values of $x_1, ..., x_{j-1}$ sampled up to iteration i and values $x_{j+1}, ..., x_N$ sampled up to iteration i.

This produces a sequence of vectors of random variables

$$(x_1^0, ..., x_N^0), (x_1^1, ..., x_N^1), ..., (x_1^k, ..., x_N^k)$$

which form a Markov chain and so the conditional distribution of any vector given all of the previous ones only depends on the vector in the previous step.

Every jump is accepted with probability 1; a consequence of the algorithm being a special case of the M-H algorithm but where conditional densities are used. This means that we are always sampling from the true posterior conditional distributions for each random variable and there is no need to reject any of the samples.

With the Gibbs sampler, no proposal distribution is required, often adding to the advantage of this method over the M-H algorithm. It also uses the most up-to-date value of each parameter, even within the same iteration, increasing the convergence speed of the chain.

Hamiltonian MCMC Both the Gibbs and M-H methods can be thought of as random walks through the parameter space, and this can be of detriment when considering problems in high dimensions. To get around this problem, Hamiltonian MCMC or Hybrid MCMC was developed by Duane *et al.* (1987) for use in lattice field theory.

The method uses ergodic dynamics rather than probability distributions to propose jumps in the chain. These dynamics are Hamiltonian dynamics, commonly used in physics to describe the evolution of a system through configuration space by using the location and momentum at time t. The Hamiltonian of a system is the sum of the potential and kinetic energies and tells us about the total energy of the system at time t. The method thus directs the chain through the parameter space under the condition that energy is conserved and thus the Hamiltonian is kept approximately constant over time.

The so-called 'leap frog' method is used to discretely simulate these continuous dynamics over time and the results are used in calculating the acceptance probability of jumps. This algorithm favours successive jumps in the same direction, allowing movement through the parameter space to happen quicker than

for an ordinary random walk.

Reversible Jump MCMC Reversible Jump MCMC (RJMCMC) as developed by Green (1995) looks to address the issue of conducting MCMC simulations with jumps in dimension, rather than sampling over a fixed number of dimensions as in the methods above. This scenario commonly arises when comparing between several proposed models, with different numbers of parameters involved. This method can be thought of as a generalization of the M-H algorithm, involving an acceptance probability with a Jacobian term.

Convergence diagnostics in MCMC methods All the above methods require some form of monitoring, in the sense that convergence of the Markov chains to the target posterior distribution is important in gaining good approximations. Thus methods have been developed to study the convergence and mixing of these chains, also ensuring that the whole of the parameter space has been sampled at some point in the simulation.

The easiest way to see convergence is to look at trace plots which monitor the jumps in the chain at each iteration of the algorithm. The x-axis of these plots is the iteration and the y-axis the sampled value of the parameter. A well-mixed chain will have explored the whole parameter space across the simulation and therefore will have a trace plot that visits all areas of the plotting space.

Numerically, convergence can be explored by looking at the acceptance rate of the algorithm. This is a measure of how often jumps are accepted, and a good acceptance rate would be somewhere between 25% and 75%, with the ideal being 50%. This is based on the acceptance probability at each step of the algorithm.

The $\hat{R}$ statistic, or Potential Scale Reduction Factor (Gelman & Rubin, 1992) can also be calculated to identify chains that have failed to converge. This is calculated for a simulation with m chains of length n after burn-in as

$$\hat{R}^2 = \frac{\mathrm{T}}{\mathrm{W}},$$

where $W = \sum_{i=1}^{m} s_i^2/m$ is the average of the within-chain variances s_i^2, and B is the variance of the means of each chain. T is then calculated as a weighted

average of B and W as $\frac{n-1}{n}W + \frac{1}{n}B$. This ratio should approach 1 as the chain converges.

When running these algorithms it is important to consider the initial values used and the proposal density chosen for the sampling of candidates. Ensuring these choices are appropriate will greatly assist the simulations and produce well-mixed, rapidly-converging chains. Autocorrelation is inherent in the resulting samples due to the nature of the Markov chain, and so thinning of observations in the chain before making calculations can produce better parameter estimates. This simply means only using every i^{th} value in the chain when calculating estimates. Samples which are highly correlated will have a lower effective sample size; the number of independent samples which would have the same precision as the total number of correlated samples present. In this case there is little movement around the parameter space. We thus have less information about the target distribution and so chains will take longer to reach and explore it.

Non-MCMC-based methods

Approximate Bayesian Computation Approximate Bayesian Computation (ABC) is a non-MCMC method, a term that was coined by Beaumont *et al.* (2002), but ideas surrounding which were first proposed by Rubin (1984). Rubin was the first to describe a sampling mechanism, which is identical to what is now known as the ABC-rejection scheme, that would give a sample from the posterior distribution of interest. Diggle & Gratton (1984) were the first to use simulations to carry out statistical inference with intractable likelihood functions, a key idea in ABC. The first ABC algorithm for inferring the posterior distribution was proposed by Tavaré *et al.* (1997). Their seminal work relates to the genealogy of DNA sequence data and they proposed the algorithm for the purpose of deciding the posterior distribution of the time to the most recent common ancestor of sampled individuals.

ABC methods are useful when the likelihood function in the problem is intractable, i.e. integration necessary for the evaluation of likelihoods is analytically difficult or impossible. It can be used when the likelihood function cannot

be written down but the data-generating mechanism is known. The methods bypass evaluation of the likelihood function, meaning that Bayesian inference can be carried out on a wider range of problems.

ABC methods approximate the likelihood function or the posterior distribution using simulations, and the outcome of these simulations are compared with the observed data. Using the ABC-rejection algorithm, this is done by initially sampling a set of parameter values from the prior distributions that define them. A data set $\hat{D}$ is then generated using the model that is defined by the sampled parameter values and subsequent statistical measures are used to compare this data set with the observed data D.

The sampled parameter values are accepted if the value of some distance measure ρ between the two data sets is less than or equal to some tolerance ϵ i.e. $\rho(D, \hat{D}) \leq \epsilon$, and are discarded otherwise. This tolerance is strictly positive and is equal to 0 if there is no discrepancy between the simulated and observed data sets. Eventually, the algorithm produces a sample of parameter values that are approximately distributed according to the posterior distribution of interest, after repeated sampling from the prior distribution and comparison of the simulated data with the observed.

When data sets with increasingly higher dimensions are used, the probability of a simulated data set being accepted typically decreases, which can lead to a reduction in efficiency of the algorithm for high-dimensional data sets. To try and reduce the effects of this, one approach is to use lower-dimensional sufficient statistics, like for example the sample mean for estimating the mean of normally distributed data in place of the observed data when running the algorithm. The sufficient statistic must contain all of the information needed to compute an estimate of the parameter of interest.

The discrepancy metric ρ then compares sufficient statistics of the simulated data $\hat{D}$ to those selected to represent observed data D. In practice though, this is often difficult to implement outside of the exponential family of distributions, as sufficient statistics must capture all information about D and $\hat{D}$. Often, summary statistics, such as moments of the observed data, are used when sufficient

statistics are not available, though a poor choice of summary statistics can lead to poor approximation of the posterior distribution (Burr & Skurikhin, 2013).

The tolerance can be set equal to 0 to obtain an exact result, though this is often extremely computationally expensive. A tolerance that is too large will lead to every point in the parameter space being accepted and will end up just yielding a copy of the prior distribution. ABC also suffers when problems include noisy data if this is not taken into account in the analyses (Schälte & Hasenauer, 2020).

Sequential Monte Carlo Sequential Monte Carlo (SMC) methods are another alternative to MCMC, proposed for Bayesian inference by Del Moral *et al.* (2006), though the first to use the term were Liu & Chen (1998). SMC involves repeated, or sequential, importance sampling from the proposal distribution with the aim of approximating the posterior distribution of interest and is well suited to running in parallel (Green & Maskell, 2016), making it a lot faster than MCMC methods.

The key idea of SMC methods is importance sampling. To sample from the target distribution $p(x)$ which is the sought posterior distribution, a proposal distribution $q(x)$ is selected which is similar to $p(x)$. Weights $w_i(x) = p_i(x)/q_i(x)$ are then assigned to each sample i taken during the algorithm, using the fact that though the distribution of $p(x)$ is only known up to a constant, its value at a point x can be calculated.

Sequential sampling from $p(x)$ is then carried out using $q(x)$ and weight $w_i(x)$ for each of the i sample values. These weights are updated in a recursive fashion with the weight in the next step being proportional to the weight in the current step. This resampling removes unimportant or low-probability values, avoiding the problem of degenerate weights.

As the algorithm goes on, in the k^{th} iteration, instead of sampling from the target distribution dependent on the current sample values only, samples are taken from the target distribution dependent on all of the previous samples, and thus an k-dimensional probability density function is needed to estimate these parameter values. For this reason, an L-kernel is used to allow sampling from

a density function of choice. This is a user-defined probability distribution that can be used to influence the efficiency of the sample (Green & Maskell, 2016). This is then multiplied by the target distribution evaluated only at the current set of sample values, with a good L-kernel being chosen to minimise the variance in any subsequent sample estimates.

SMC can be used in combination with ABC to produce independent samples, an advantage over ABC-MCMC methods. The tolerance levels used in ABC are also not required to be specified when in combination with SMC, as they are adjusted along with the sequence (Del Moral *et al.*, 2012).

Variational Bayes Variational inference, and in particular Variational Bayes (VB) is another broad category of non-MCMC method that is widely used. Models including latent variables, unobserved variables that are often inferred from those variables that are observed, make particular use of these methods. These methods are again useful when it is not feasible to obtain the posterior distribution analytically, however where MCMC methods provide an approximation to the posterior using repeated sampling, VB can produce an exact analytical approximation of the posterior. There is however more work involved in deriving the equations used to update the parameters in VB compared to those used for MCMC simulations.

The first variational procedure based around estimation in a neural network was studied by Peterson & Anderson (1987), and Parisi (1998) contributed insights on this method using statistical mechanics, leading to variational inference on a wider class of models. Hinton & van Camp (1993) proposed in parallel a variational algorithm for a different neural network, followed by the work of Neal & Hinton (1993) and Neal & Hinton (1998), making connections to the Expectation Maximization theorem (Dempster *et al.*, 1977). This in turn led to several variational inference algorithms for other classes of models.

The idea behind VB is to get as close an approximation as possible to the posterior distribution using an intermediate distribution $Q(\theta)$. $Q(\theta)$ is an arbitrary distribution from some family of distributions chosen to represent the unobserved and latent variables in the model and its parameters can be fine-tuned until the

closest approximation to the posterior is found.

Tuning the parameters so that the Kullback-Leiber divergence is minimised will give the closest approximation to the posterior distribution. This optimization is often done using some iterative optimisation procedure such as the Expectation Maximisation algorithm (Dempster *et al.*, 1977). Although other measures of dissimilarity are used in VB, the Kullback-Liebler is the most common as the minimization of this quantity is achieved easily.

3.1.3 Application of Bayesian methods to biological applications, cancer research and tumour growth

Bayesian methods are a popular way to analyse biological data and are frequently used on to estimate parameters in biological systems, especially in relation to cancer, tumour growth and cell motility. It is beyond the scope of this introduction to include all such works using Bayesian analysis, but some examples from recent literature where the methods outlined above are used are given here.

In a general biological context, Bayesian methods are widely used to study a variety of topics. MCMC methods are particularly common and are being utilized in the fields of genetics, epidemiology, and evolutionary biology, to name a few.

In the field of genetics, Husmeier & McGuire (2002) use MCMC methods to detect recombination in DNA sequences, important for understanding genetic diversity and how it comes about. They use MCMC simulations on phylogenetic tree topologies at various states and for windows of nucleotide sequences to identify the locations of recombinant DNA. Li *et al.* (2011) similarly tackle the problem of detection, this time looking for repetition in DNA sequences which can uncover biological structure and function of proteins coded for by these sequences. They use an MCMC algorithm, using both Gibbs sampling and the Metropolis-Hastings algorithm, to estimate parameter values to define the location and structure of the repeated segments. An adaptive MCMC algorithm is outlined by Baele *et al.* (2017) for studying phylogenetic trees, providing novel ways to cope with the large data sets and numbers of parameters to be estimated

in bioinformatics data sets used.

MCMC methods are also used in imaging applications, a growing area of research with important implications for medicine. Diffusion magnetic resonance imaging analysis is studied in Harms & Roebroeck (2018), where they show that adaptive MCMC can increase MCMC performance when matching imaging signals to image data, providing a full posterior for parameters rather than the typically used maximum likelihood estimates. Ihsani *et al.* (2018) have used MCMC to estimate kinetic parameters in positron emission tomography imaging to quantify the impact of a stimulus, for example the degree of ischaemia in heart tissue.

In the context of epidemiology, Cauchemez *et al.* (2004) use MCMC methods to investigate transmission of influenza within households, estimating the duration of the infectious period, the instantaneous risk of infection and the dependence of this risk on the density of infected individuals in the household. MacLehose *et al.* (2007) look at highly correlated exposures to, for example, multiple pathogens or pesticides. The effect of multiple exposures is of interest and often frequentist regression is used to try and estimate relevant coefficients. In this work, MacLehose *et al.* (2007) use Bayesian hierarchical models and MCMC methods to circumvent the problems of poor convergence and loss of information, thus giving more accurate and reliable parameter estimates.

The pharmaceutical industry also makes use of MCMC methods during drug discovery and development. Trägårdh *et al.* (2016) applied MCMC algorithms to the study non-linear systems of ODEs which define pharmacokinetic and pharmacodynamic models for various drugs. MCMC is of particular use here as most of the required parameter values cannot be written in a closed analytic form. Bois *et al.* (2020) also consider pharmacokinetic models and show that the use of tempered MCMC improves mixing of chains and is able to deal well with multi-modal posteriors that can be common in this type of modelling. The problem of determining the maximum tolerated dose of a drug during a clinical trial is studied by Ye *et al.* (2020), considering a single toxicity response to the drug where this toxicity increases with dosage.

More specific to the work considered in this chapter is the use of Bayesian methods as applied to cancer and cell motility. The mathematical oncology roadmap produced by Rockne *et al.* (2019) outlines methods currently in use for studying mathematical modelling of cancer, and several Bayesian techniques are reported on in this document, including Bayesian model selection and applications of Bayesian ideas to deep learning techniques.

Ellis *et al.* (2015) provide an overview of current challenges in glioblastoma and outline how Bayesian networks, which are probabilistic graphical models allowing inferences to be made around model variables, are being used in favour of artificial neural networks to model interactions, pathways and processes involved in tumour spread and growth. This is appropriate given that data can be scarce and the opinion of clinicians should be taken into account as prior information in clinical prediction work such as this.

Lipková *et al.* (2019) use a Bayesian framework based on Transitional MCMC to look at personalized radiotherapy treatment for patients with glioblastoma tumours, calibrating the model to the data and using Bayesian Inference to predict individual tumour cell density. They make use of a deterministic partial differential equation (PDE) model which uses Fisher-Kolmogorov equations to model tumour density and then create a stochastic imaging model to relate this density to observations from MRI imaging data. The stochastic component uses the Bernoulli distribution to model probabilities of observing certain imaging signals with the simulated cell densities from the PDE.

RJMCMC is employed in Pravitasari *et al.* (2019) for optimizing image segmentation in MRI scans to diagnose brain cancers. They use the dimension-jumping ability of the method to select the optimum number of clusters of tumour locations. Lê *et al.* (2015) use Gaussian Process Hamiltonian Monte Carlo to personalize parameters in a typically used reaction-diffusion model for tumour growth, including a logistic proliferation term, for patients with glioblastoma tumours.

Kursawe *et al.* (2018) use an Approximate Bayesian inference scheme to show that parameters in proliferative models of epithelial cell growth can be inferred

from imaging data and their uncertainty quantified. They use a simplified vertex model, numerically solving equations for position of a vertex at time i and energy associated with each tissue using a forward Euler scheme. Inference of parameters is carried out using a comprehensive list of summary statistics, as is standard for an ABC scheme, including average cell perimeter, correlation between areas of adjacent cells and average cell elongation.

In Toni *et al.* (2009) an ABC-SMC scheme for model selection in dynamical systems is developed and the use of this scheme for tumour modelling is later confirmed in da Costa *et al.* (2018) where it is shown that the scheme can accurately select the correct models with accurate parameter estimates for patients both receiving chemotherapy and not. The models considered in this work are mostly systems of ordinary differential equations (ODEs), including a Gompertz model and an exponential model, with one coupled system of reaction-diffusion equations that include both ODEs and a PDE. All of the models include variables that describe the number of cells and the concentration of drug when chemotherapy was administered, with parameters including cell growth rate, cell reduction after chemotherapy and drug decay rate being estimated using Bayesian inference.

SMC is used also in Ogundijo & Wang (2018) and Ogundijo *et al.* (2019) to study tumour heterogeneity. In the first instance (Ogundijo & Wang, 2018), heterogeneity in tumours is characterized, as defined by the haplotype of cells, and SMC is thus employed to estimate these latent haplotypes by characterizing their types and the proportions of them present in patient samples. In Ogundijo *et al.* (2019), the algorithm is employed to estimate the number of subpopulations of different cells, or subclones, present in a heterogeneous tumour using mutation data.

In Matsutani *et al.* (2019), a novel method for estimating the number of mutation signatures in cancer is presented through the use of Variational Bayesian inference, where lower bounds employed in the method are used to find a plausible number of mutation patterns.

Bayesian optimization and calibration methods are also being applied to tu-

mour modelling. Hawkins-Daarud *et al.* (2013) present a Bayesian framework for calibration, validation and uncertainty quantification of tumour growth models, demonstrated on initial-boundary value problems for concentrations of tumour cells and nutrients present in tissues. Collis *et al.* (2017) produce a tutorial on the same topic, influenced by the work of Hawkins-Daarud *et al.* (2013).

3.2 Analysis

This chapter will look again at the tracking data studied in chapter 2, this time analysing it from a Bayesian perspective. The goal is to compare the two types of analysis, outlining and considering the advantages and disadvantages of each of the methods used.

An overview of the Bayesian methods used will be provided in the first instance, specifically the use of the Gibbs sampler detailed above, chosen because the likelihood function is easily obtained for the models considered. The availability of software and tools for interpretation of MCMC simulation outputs was also anticipated to prove useful for comparing estimates across frequentist and Bayesian analyses.

A presentation of the results of using different models and priors in analyses of the *in silico* and experimental data sets will then follow. Model selection is explored before advantages and disadvantages of both methods are considered and discussion and subsequent conclusions are made.

3.2.1 Overview of approach

Using the framework for 3D cell tracking data outlined in chapters 1 and 2, the Bayesian approach is now considered in contrast to the frequentist one previously used. It is reasonable that the frequentist analysis in the framework could be replaced by a Bayesian one and so in order to compare and contrast with the classical methods, the previous analysis is repeated using Bayesian ideas and estimates of S and P are obtained.

In this approach, rather than using the SDE for the PRW model as the

governing model and directly applying Bayesian analyses to the resulting cell tracks, we use the summary statistics $\ln(VACF)$ and $RMSS$ and fit various regression models with AR processes of different orders to them. This means that we are able to estimate the parameters as in the frequentist framework, but here we don't test the goodness-of-fit of the PRW model to the tracking data.

For estimating parameter S independently, the $RMSS$ time series data is used, and for estimating S and P together, the $\ln(VACF)$ data is used. The proposed model and prior distributions for parameters are first defined and then MCMC simulations are run using the R package rjags (Plummer, 2019). This package uses a Gibbs sampler, with necessary adjustment of simulation parameters, such as number of chains used and length of burn-in period, to allow parameter estimates to be taken once chains are well-mixed and have converged.

Initial parameter values in the simulations were chosen based on knowledge of parameter values from experiments or the frequentist analyses. The number of chains, iterations and burn-in were held constant within the analyses of each data set, changing between data sets where necessary and as detailed below.

Five data sets were considered in the analyses, chosen as they have been studied in the frequentist analyses, and include two *in silico* data sets with $S = 1$, $P = 1$, $dt = 0.01$, referred to henceforth as 11001, and $S = 25$, $P = 0.1$, $dt = 0.05$, referred to as 2501, and the three experimental data sets from the control spheroids, Spheroid 1, Spheroid 2 and Spheroid 3.

Four analytical cases are considered here to explore different ways to estimate parameters and the effect of different priors on the corresponding estimates. In sections 3.2.2 and 3.2.3, S and P are estimated independently of each other, as in the frequentist analysis, and in sections 3.2.4 and 3.2.5 a different approach is proposed for estimating S and P simultaneously. Convergence was checked in all cases by looking at trace plots, densities and the $\hat{R}$ statistic (Gelman & Rubin, 1992), though these outputs are not always shown for the sake of brevity. Visualisations are presented along with output and estimates in section 3.2.5, where R package runjags (Denwood, 2016) is used to ensure convergence of chains.

For convenience, the estimates from the frequentist framework are replicated

	$\hat{\mathbf{S}}$	$\hat{\mathbf{P}}$
11001	0.9978 [0.9892, 1.0064]	0.9893 [0.9473, 1.0352]
2501	25.0458 [24.9091, 25.1825]	0.0996 [0.0978, 0.1015]
Spheroid 1	27.3137 μm/h [25.2892, 29.3382]	0.0863 h [0.0697, 0.1130]
Spheroid 2	26.9272 μm/h [25.9613, 27.8930],	0.0789 h [0.0677, 0.0946]
Spheroid 3	28.0600 μm/h [27.3979, 28.7222],	0.0976 h [0.0804, 0.1241]

Table 3.1: Parameter estimates for the 5 data sets to be considered by Bayesian analyses, taken from the frequentist framework analyses in chapter 2. Estimates are accompanied by 95% confidence intervals.

in table 3.1 for the 5 data sets considered in the following analyses.

3.2.2 Estimating S alone

Intending to replicate results from the frequentist approach, the first estimates of S were calculated independently using the $RMSS$ time series data. This meant assuming that S was the stationary mean of an AR(1) process with correlation coefficient ϕ, some constant μ and errors $\epsilon_t \sim N(0, \sigma^2)$ i.e.

$$\begin{aligned} & Y_t = \mu + \phi Y_{t-1} + \epsilon_t \\ \implies & E[Y_t] = E[\mu] + E[\phi Y_{t-1}] + E[\epsilon_t] \\ \implies & E[Y_t] = \mu + \phi E[Y_t] \\ \implies & E[Y_t] = \frac{\mu}{1-\phi} = S, \end{aligned}$$

so $\mu = S(1-\phi)$ and the AR(1) model in terms of the variables in the data sets becomes

$$RMSS_i = S(1-\phi) + \phi\, RMSS_{i-1} + \epsilon_i,$$

for time step i and observations of $RMSS$ at each of these time steps. The correlation here is seen between the subsequent observations in the time series.

The precision parameter $\tau = 1/\text{sd.obs}^2$, where sd.obs is the standard deviation of the distribution of the response variable in the model (here $RMSS$ and later $\ln(VACF)$), will be used throughout simulations as an alternative way to

parametrize the variance. It is standard practice to define normal distributions in JAGS models using precision rather than variance. Though priors for τ are defined and required by `rjags`, sd.obs is monitored in the simulations for ease of understanding.

Priors for S were given as gamma distributions in all cases due to S being a positive value and the belief that its distribution would be skewed rather than centred around one particular value. The prior for S in *in silico* data sets was informed by the known value of S when the data was simulated in that the mean of the gamma distribution was set to this known value. Since the scale parameter is equal to 1, the mean of the distribution is simply equal to the shape parameter. For experimental data, the prior was informed by estimates from the frequentist framework and the experimentalists. The prior mean was thus taken to be 27 in all cases, this being close to the average of all 3 experimental estimates of S previously calculated, and again the scale parameter was set to 1.

The prior for correlation coefficient ϕ is left uninformative, and thus uniform over the interval $[-1, 1]$ as there is no prior knowledge of the value of this correlation. Initial values were chosen either arbitrarily, or to be close to known values, i.e. $S = 1$ when this is known in the *in silico* case, to aid faster convergence of chains. The precision parameter τ was given a tight gamma prior such that the prior on the variance, the inverse of the precision, would be uninformative.

The JAGS model code for these simulations can be found in Appendix B.1 and the code for running the MCMC can be found in Appendix C.1. The results of this first analysis are shown in tables 3.2 for *in silico* data sets and 3.3 for experimental data.

These results show that in all cases, point estimates of $\hat{S}$ are close to what we expect and in the *in silico* cases the known values of S are within the credibility intervals. These intervals are quite wide for the experimental data sets, but the estimates are close to what experimentalists had estimated S to be, and certainly the values estimated by the experimentalists lie within the credibility intervals. We see varying degrees of correlation being estimated in all of the simulated cases, though all estimate the posterior mean of $\hat{\phi}$ to be greater than 0.45, with

	11001	**2501**
Sample size	1001	101
S **prior**	Gamma(1,1)	Gamma(25,1)
ϕ **prior**	Unif(-1,1)	Unif(-1,1)
τ **prior**	Gamma(0.001, 0.001)	Gamma(0.001, 0.001)
Initial values (ϕ, S, τ)	0.5, 1, 1	0.5, 25, 1
Chains	2	2
Iterations, burn-in	50000, 37500	50000, 37500
$\hat{S}$ **mean**	0.9964	25.0428
$\hat{S}$ **95% CI**	[0.9265, 1.0657]	[24.8858, 25.1999]
$\hat{\phi}$ **mean**	0.9848	0.4599
$\hat{\phi}$ **95% CI**	[0.9688, 0.9997]	[0.2755, 0.6455]
$\hat{\mu}$ **mean**	0.0151	13.5247
$\hat{\mu}$ **95% CI**	[0.0003, 0.0311]	[8.8861, 18.1511]

Table 3.2: Results of MCMC simulations on *in silico* data sets for parameter estimates in the 'Estimating S alone' case, detailing the priors, parameter estimates and 95% credibility intervals (CI) for estimates. 11001 refers to the data set where $S = 1$, $P = 1$ and 2501 to the data set where $S = 25$, $P = 0.1$.

	Spheroid 1	**Spheroid 2**	**Spheroid 3**
Sample size	149	93	78
$\hat{S}$ **prior**	Gamma(25,1)	Gamma(25,1)	Gamma(25,1)
$\hat{\phi}$ **prior**	Unif(-1,1)	Unif(-1,1)	Unif(-1,1)
τ **prior**	Gamma(0.001, 0.001)	Gamma(0.001, 0.001)	Gamma(0.001, 0.001)
Initial values ($\hat{\phi}$, $\hat{S}$, τ)	0.5, 27, 1	0.5, 27, 1	0.5, 27, 1
Chains	2	2	2
Iterations, burn-in	50000, 37500	50000, 37500	50000, 37500
$\hat{S}$ **mean** (μ**m/h**)	26.5262	26.4246	27.9767
$\hat{S}$ **95% CI**	[23.7393, 28.8190]	[24.2216, 27.9875]	[27.2197, 28.7082]
$\hat{\phi}$ **mean**	0.8338	0.7715	0.4824
$\hat{\phi}$ **95% CI**	[0.7409, 0.9282]	[0.6094, 0.9385]	[0.2785, 0.6923]
$\hat{\mu}$ **mean**	4.4243	6.0600	14.4838
$\hat{\mu}$ **95% CI**	[1.8164, 6.9864]	[1.5617, 10.4581]	[8.5802, 20.2120]

Table 3.3: Results of MCMC simulations on experimental data for parameter estimates in the 'Estimating S alone' case, detailing the priors, parameter estimates and 95% credibility intervals (CI) for estimates.

95% credibility intervals containing only positive values.

Comparing to the frequentist framework, point estimates of $\hat{S}$ are similar to those obtained previously, and confidence intervals seem to be of similar widths to credibility intervals here, with the exception of Spheroid 1 where the credibility interval is wider than the confidence interval in the frequentist framework. The point estimates of, and intervals for, $\hat{S}$ in the 2501 *in silico* cases are remarkably similar in the frequentist and Bayesian analyses. The point estimate of $\hat{S}$ in the 11001 *in silico* case is closer to the true value in the Bayesian analysis than in the frequentist, though the credibility interval is much wider than the confidence interval. This is what we would expect though, given that the priors were informed by knowing the true values of S or using previous estimates.

3.2.3 Estimating P alone

We now estimate P using the same method as in the frequentist approach, with $\ln(VACF)$ data being used to fit a regression model with correlated errors and the estimate of P coming from the negative reciprocal of the slope coefficient of this model. The cut-off as used in the frequentist framework is also used here when choosing which subset of the data to use in estimating P, hence why the sample sizes are greatly reduced.

Taking logs of the expression for $VACF(t)$ in equation 1.6, we get

$$\ln(VACF) = \ln(S^2) - \frac{1}{P}t,$$

which clearly is a straight line with slope $-1/P$ and intercept $\ln(S^2)$. The correlation here is seen in the errors and they are assumed to follow an AR(1) process. The model being fitted is thus

$$\ln(VACF)_i = a - \frac{1}{P}t_i + \epsilon_i,$$

for time step i and observations of $\ln(VACF)$ at the i^{th} time step, where $\epsilon_i = \phi\,\epsilon_{i-1} + u_i$ and $u_i \sim N(0, 1/\tau)$.

Two cases are reported in this analysis. One where intercept a is given an uniform prior, disregarding any dependence on S, and one where the prior for a

is informed by frequentist estimates of the intercept $\ln(S^2)$ and their confidence intervals. Priors for ϕ and τ are the same as those given in section 3.2.2.

In the first instance the priors for a were chosen to be uniform to look at what a flat prior could tell us about the intercept. The range of this uniform distribution is determined in each case from confidence intervals around the frequentist estimates of the intercept. Appropriate ranges for the uniform priors were selected in each case with the centre of the interval being roughly the frequentist estimate and the range covering the whole of the confidence interval with some spread added on either side to allow for extra variability. The results of these simulations are given in tables 3.4 and 3.5.

In the second case the priors for a were given as normal distributions, informed again by the frequentist point estimates. The results of these analyses are shown in tables 3.6 and 3.7.

For the experimental data sets and the *in silico case* 2501 which mimics the experimental parameters, uniform priors were used for P. These ranged between 0.06 and 0.4, the frequentist framework leading to the belief that P for experimental data sets is around 0.1, though values as high as 0.3 had been observed in previous runs of the Bayesian analyses. The flat nature of this prior across the chosen interval is also important as we are less certain about the value of P than of S. A gamma distribution is used as the prior for P in the *in silico* case where $S = P = 1$ as we know the true value of P is 1, meaning the prior is given a mean value of 1. JAGS model code for these simulations can be found in Appendix B.2.

In the case where the prior for a is uniform, point estimates of $\hat{P}$ are reasonably consistent across all experimental data sets, between 0.18 h and 0.21 h, though credibility intervals are very wide. This is dissimilar to the frequentist analyses where confidence intervals are narrower, likely due to the priors here being uninformative. In the *in silico* cases the point estimates of $\hat{P}$ are as close to the true values as in the frequentist analyses, though the credibility interval in the 2501 case is again wider than the confidence interval. The credibility interval for the 11001 case is narrower than the frequentist confidence interval,

	11001	**2501**
Sample size	59	5
a **prior**	Unif(-1,1)	Unif(5,9)
P **prior**	Gamma(4,0.25)	Unif(0.06,0.4)
ϕ **prior**	Unif(-1,1)	Unif(-1,1)
τ **prior**	Gamma(0.001, 0.001)	Gamma(0.001, 0.001)
Initial values (a, P, ϕ, τ)	0, 1, 0.5, 1	7, 0.1, 0.5, 1
Chains	2	2
Iterations, burn-in	10,000, 5,000	10,000, 5,000
$\hat{a}$ **mean**	-0.0021	6.3678
$\hat{a}$ **95% CI**	[-0.0079, 0.0038]	[5.4871, 6.7638]
$\hat{S} = e^{a/2}$	0.9990	24.1407
$\hat{P}$ **mean**	1.0176	0.1137
$\hat{P}$ **95% CI**	[0.9998, 1.0361]	[0.0852, 0.2987]
$\hat{\phi}$ **mean**	0.8826	0.1789
$\hat{\phi}$ **95% CI**	[0.7714, 0.9962]	[-0.9333, 0.9592]

Table 3.4: Results of MCMC simulations for parameter estimates in the 'Estimating P alone' case on *in silico* data sets with AR(1) errors, detailing the priors, parameter estimates and 95% credibility intervals (CI) for estimates. 11001 refers to the data set where $S = 1$, $P = 1$ and 2501 to the data set where $S = 25$, $P = 0.1$.

	Spheroid 1	**Spheroid 2**	**Spheroid 3**
Sample size	5	6	5
a **prior**	Unif(5,9)	Unif(5,9)	Unif(5,9)
P **prior**	Unif(0.06,0.4)	Unif(0.06,0.4)	Unif(0.06,0.4)
ϕ **prior**	Unif(-1,1)	Unif(-1,1)	Unif(-1,1)
τ **prior**	Gamma(0.001, 0.001)	Gamma(0.001, 0.001)	Gamma(0.001, 0.001)
Initial values (a, P, ϕ, τ)	7, 0.08, 0.5, 1	7, 0.08, 0.5, 1	7, 0.08, 0.5, 1
Chains	2	2	2
Iterations, burn-in	50,000, 25,000	50,000, 25,000	50,000, 25,000
$\hat{a}$ **mean**	6.7120	6.0989	6.6645
$\hat{a}$ **95% CI**	[5.7977, 7.8732]	[5.7421, 7.5773]	[6.1508, 7.6325]
$\hat{S} = e^{a/2}$ **(**μ**m/h)**	28.6743	21.1037	28.0013
$\hat{P}$ **mean (h)**	0.2116	0.1919	0.1846
$\hat{P}$ **95% CI**	[0.0732, 0.3878]	[0.0709, 0.3975]	[0.0771, 0.3843]
$\hat{\phi}$ **mean**	0.4763	0.5137	0.2312
$\hat{\phi}$ **95% CI**	[-0.8743, 0.9897]	[-0.8617, 0.9921]	[-0.9380, 0.9851]

Table 3.5: Results of MCMC simulations for parameter estimates in the 'Estimating P alone' case on experimental data sets with AR(1) errors, detailing the priors, parameter estimates and 95% credibility intervals (CI) for estimates.

	11001	**2501**
Sample size	59	5
a **prior**	N(0,0.01)	N(7,1)
P **prior**	Gamma(4,0.25)	Unif(0.06,0.4)
ϕ **prior**	Unif(-1,1)	Unif(-1,1)
τ **prior**	Gamma(0.001, 0.001)	Gamma(0.001, 0.001)
Initial values (a, P, ϕ, τ)	0, 1, 0.5, 1	7, 0.1, 0.5, 1
Chains	2	2
Iterations, burn-in	10,000, 5,000	50,000, 25,000
$\hat{a}$ **mean**	-0.0022	6.4294
$\hat{a}$ **95% CI**	[-0.0080, 0.0038]	[5.9032, 6.8033]
$\hat{S} = e^{a/2}$	0.9989	24.8958
$\hat{P}$ **mean**	1.0180	0.1045
$\hat{P}$ **95% CI**	[1.0000, 1.0359]	[0.0810, 0.1661]
$\hat{\phi}$ **mean**	0.8868	0.1220
$\hat{\phi}$ **95% CI**	[0.7723, 0.9970]	[-0.9428, 0.9519]

Table 3.6: Results of MCMC simulations for parameter estimates in the 'Estimating P alone' case on *in silico* data sets with AR(1) errors, detailing the priors, parameter estimates and 95% credibility intervals (CI) for estimates. 11001 refers to the data set where $S = 1$, $P = 1$ and 2501 to the data set where $S = 25$, $P = 0.1$.

	Spheroid 1	**Spheroid 2**	**Spheroid 3**
Sample size	5	6	5
a **prior**	N(7,1)	N(7,1)	N(7,1)
P **prior**	Unif(0.06,0.4)	Unif(0.06,0.4)	Unif(0.06,0.4)
ϕ **prior**	Unif(-1,1)	Unif(-1,1)	Unif(-1,1)
τ **prior**	Gamma(0.001, 0.001)	Gamma(0.001, 0.001)	Gamma(0.001, 0.001)
Initial values (a, P, ϕ, τ)	7, 0.08, 0.5, 1	7, 0.08, 0.5, 1	7, 0.08, 0.5, 1
Chains	2	2	2
Iterations, burn-in	50,000, 25,000	50,000, 25,000	50,000, 25,000
$\hat{a}$ **mean**	6.7370	6.3791	6.7661
$\hat{a}$ **95% CI**	[5.9294, 7.7908]	[5.2810, 7.4609]	[5.8532, 7.5297]
$\hat{S} = e^{a/2}$ **(**μ**m/h)**	29.0349	24.2775	29.4605
$\hat{P}$ **mean (h)**	0.2081	0.1613	0.1661
$\hat{P}$ **95% CI**	[0.0749, 0.3863]	[0.0690, 0.3686]	[0.0781, 0.3778]
$\hat{\phi}$ **mean**	0.4755	0.3390	0.1223
$\hat{\phi}$ **95% CI**	[-0.8711, 0.9906]	[-0.9162, 0.9898]	[-0.9559, 0.9816]

Table 3.7: Results of MCMC simulations for parameter estimates in the 'Estimating P alone' case on experimental data sets with AR(1) errors, detailing the priors, parameter estimates and 95% credibility intervals (CI) for estimates.

and this data set is substantially larger than the others, perhaps suggesting that a larger sample size in other data sets could give credibility intervals narrower than confidence intervals if such data was available.

When the prior for a is normal, the point estimates of $\hat{P}$ are consistent in the experimental data sets in the cases of Spheroid 2 and Spheroid 3, but overall the estimates are again larger than what is seen in the frequentist analyses with much wider credibility intervals. The *in silico* estimates are fairly close to known values, but the credibility interval in the 11001 case only just contains the true value, $P = 1$.

Using the point estimates of $\hat{a}$ given by the Bayesian analysis, it is possible to calculate estimates of S using the fact that $\hat{a} = \ln(\hat{S}^2) \implies \hat{S} = e^{\hat{a}/2}$. Doing so gives mixed results, and in both the analyses with uniform and normal priors for a, we see very good estimates in the 11001 case and good estimates in the 2501 case when compared with the known values of S. We see very variable estimates from the experimental data sets, with S being estimated as higher overall when the prior for a is normal.

It is also of note that point estimates of correlation coefficient $\hat{\phi}$ are fairly consistent across the different cases of a prior, but are vastly different between data sets.

3.2.4 Estimating P and S simultaneously

We now consider S as being part of the intercept and thus estimate S and P simultaneously, something which was not done in the frequentist framework. We only use the $\ln(VACF)$ data in this section and don't include any information from the $RMSS$ data. Again we assume the errors follow an AR(1) process, the difference here being that the dependence on S is made explicit and priors are defined for S rather than some intercept a. The model being fitted here is then

$$\ln(VACF)_i = \ln(S^2) - \frac{1}{P}t_i + \epsilon_i, \tag{3.4}$$

for time step i and observations of $\ln(VACF)$ at the i^{th} time step, where $\epsilon_i = \phi\,\epsilon_{i-1} + u_i$ and $u_i \sim N(0, 1/\tau)$.

	11001	**2501**
Sample size	59	5
S **prior**	Gamma(4,0.25)	Gamma(25,1)
P **prior**	Gamma(4,0.25)	Unif(0.06,0.4)
ϕ **prior**	Unif(-1,1)	Unif(-1,1)
τ **prior**	Gamma(0.001, 0.001)	Gamma(0.001, 0.001)
Initial values (S, P, ϕ, τ)	1,1,0.5,1	25, 0.1, 0.5, 1
Chains	2	2
Iterations, burn-in	10,000, 5,000	10,000, 5,000
$\hat{S}$ **mean**	0.9990	25.0800
$\hat{S}$ **95% CI**	[0.9961, 1.0019]	[20.6029, 29.1533]
$\hat{P}$ **mean**	1.0173	0.1023
$\hat{P}$ **95% CI**	[0.9996, 1.0360]	[0.0829, 0.1388]
$\hat{\phi}$ **mean**	0.8804	0.0678
$\hat{\phi}$ **95% CI**	[0.7681, 0.9852]	[-0.9494, 0.9457]

Table 3.8: Results of MCMC simulations on *in silico* data sets for parameter estimates in the 'Estimating P and S simultaneously' case with AR(1) errors, detailing the priors, parameter estimates and 95% credibility intervals (CI) for estimates. 11001 refers to the data set where $S = 1$, $P = 1$ and 2501 to the data set where $S = 25$, $P = 0.1$.

Parameter S was given the same priors as those in section 3.2.2, informed by the frequentist estimates. $\ln(VACF)$ data used was cut at the same point as above to create a subset of data and priors for P in this analysis are the same as those in section 3.2.3. The results can be seen in tables 3.8 for *in silico* data and and 3.9 for experimental data.

Point estimates of $\hat{P}$ here are much higher than frequentist estimates in the experimental cases though with very wide credibility intervals, similar to what was seen in section 3.2.3. The point estimates of $\hat{P}$ in the *in silico* data sets are close to the true values and the credibility intervals include these true values.

	Spheroid 1	**Spheroid 2**	**Spheroid 3**
Sample size	5	6	5
S **prior**	Gamma(25,1)	Gamma(25,1)	Gamma(25,1)
P **prior**	Unif(0.06,0.4)	Unif(0.06,0.4)	Unif(0.06,0.4)
ϕ **prior**	Unif(-1,1)	Unif(-1,1)	Unif(-1,1)
τ **prior**	Gamma(0.001, 0.001)	Gamma(0.001, 0.001)	Gamma(0.001, 0.001)
Initial values (S, P, ϕ, τ)	27, 0.08, 0.5, 1	27, 0.08, 0.5, 1	27, 0.08, 0.5, 1
Chains	2	2	2
Iterations, burn-in	10,000, 5,000	10,000, 5,000	10,000, 5,000
$\hat{S}$ **mean** (μ**m/h**)	25.2142	23.5377	25.4714
$\hat{S}$ **95% CI**	[19.6853, 34.8172]	[15.5820, 35.3656]	[18.7650, 36.0225]
$\hat{P}$ **mean (h)**	0.2535	0.1682	0.2188
$\hat{P}$ **95% CI**	[0.1073, 0.3903]	[0.0769, 0.3579]	[0.0931, 0.3869]
$\hat{\phi}$ **mean**	0.7429	0.4801	0.4735
$\hat{\phi}$ **95% CI**	[-0.1744, 0.9930]	[-0.8443, 0.9905]	[-0.9150, 0.9910]

Table 3.9: Results of MCMC simulations on experimental data sets for parameter estimates in the 'Estimating P and S simultaneously' case with AR(1) errors, detailing the priors, parameter estimates and 95% credibility intervals (CI) for estimates.

The $\hat{S}$ point estimates are similarly close to the true values in the *in silico* cases, with the credibility interval in the 11001 case being incredibly narrow. The credibility interval in the 2501 case is however quite wide, as are all of the intervals surrounding the experimental $\hat{S}$ estimates. There is a high degree of positive correlation in all data sets except in the 2501 case, though the confidence intervals suggest that this is only significant in the 11001 case.

3.2.5 Estimating P and S simultaneously - S prior informed

The final set of estimates were calculated with the same uniform priors for P as for the analyses in section 3.2.4, and informed priors for S based on estimates of S from section 3.2.2. These were formed as gamma distributions, Gamma(a, b), with parameters a and b defined by the posterior mean and variance of $\hat{S}$, using the corresponding estimates for each data set from section 3.2.2. The mean and variance needed for the informative priors are calculated using the set of simultaneous equations,

$$E\left[\hat{S}\right] = a/b,$$
$$\text{Var}\left[\hat{S}\right] = a/b^2.$$

Priors for ϕ are now given beta distributions as we expect positive correlation. Both parameters defining this beta distribution are chosen as 1.5 so that there is a preference for mid-range values of ϕ but this is not too stark.

The model being fitted here is given by equation 3.4, and with the exception of the ϕ and S priors, all priors remain as in section 3.2.4. The results of these simulations are shown in tables 3.10 and 3.11 and example visual output for the 11001 *in silico* case and the Spheroid 2 experimental case are shown in figures 3.1 and 3.2, respectively. JAGS model code can be found in Appendix B.3 and example MCMC simulation code in Appendix C.2.

For this analytical case, the R package runjags (Denwood, 2016) was used to run the MCMC simulations, this allowing use of the `autorun.jags` function which runs the simulations until convergence is reached, evidenced by an $\hat{R}$ value

	11001	**2501**
Sample size	59	5
S **prior**	Gamma(77,78)	Gamma(100183,4000)
P **prior**	Gamma(4,0.25)	Unif(0.06,0.4)
ϕ **prior**	Beta(2,2)	Beta(1.5,1.5)
τ **prior**	Gamma(0.001, 0.001)	Gamma(0.001, 0.001)
Initial values (S, P, ϕ, τ)	1, 1, 0.5, 1	25, 0.1, 0.5, 1
Chains	4	4
Iterations, burn-in	100,000, 100,000	100,000, 100,000
$\hat{S}$ **mean**	0.9992	25.0477
$\hat{S}$ **95% CI**	[0.9963, 1.0020]	[24.8869, 25.1980]
$\hat{P}$ **mean**	1.0164	0.1004
$\hat{P}$ **95% CI**	[0.9990, 1.0339]	[0.0903, 0.1103]
$\hat{\phi}$ **mean**	0.8576	0.4143
$\hat{\phi}$ **95% CI**	[0.7539, 0.9548]	[0.0157, 0.8403]

Table 3.10: Results of MCMC simulations on *in silico* data sets for parameter estimates in the 'Estimating P and S simultaneously - S prior informed' case with AR(1) errors, detailing the priors, parameter estimates and 95% credibility intervals (CI) for estimates. 11001 refers to the data set where $S = 1$, $P = 1$ and 2501 to the data set where $S = 25$, $P = 0.1$.

of less than 1.05 for all parameters. This was chosen to ensure that chains had converged and were well-mixed in this most crucial analysis.

In this final analysis, *in silico* point estimates of $\hat{S}$ and $\hat{P}$ are very close to the known values, with credibility intervals including these true values and being narrow. This indicates that these estimates are just as reliable as in the frequentist analyses, even when the sample size is as small as 5 in the 2501 case and with the flat uniform prior for P.

Experimental point estimates of $\hat{S}$ in this analysis are close to what we observe in the frequentist framework though $\hat{P}$ values are higher than what was

	Spheroid 1	**Spheroid 2**	**Spheroid 3**
Sample size	5	6	5
S **prior**	Gamma(433,16)	Gamma(564,21)	Gamma(5466,195)
P **prior**	Unif(0.06,0.4)	Unif(0.06,0.4)	Unif(0.06,0.4)
ϕ **prior**	Beta(1.5,1.5)	Beta(1.5,1.5)	Beta(1.5,1.5)
τ **prior**	Gamma(0.001, 0.001)	Gamma(0.001, 0.001)	Gamma(0.001, 0.001)
Initial values (S, P, ϕ, τ)	27, 0.08, 0.5, 1	27, 0.08, 0.5, 1	27, 0.08, 0.5, 1
Chains	4	4	4
Iterations, burn-in	100,000, 100,000	100,000, 100,000	100,000, 100,000
$\hat{S}$ **mean** (μ/**h**)	26.7482	26.7746	28.0156
$\hat{S}$ **95% CI**	[24.2668, 29.2366]	[24.6016, 29.0240]	[27.2725, 28.7554]
$\hat{P}$ **mean (h)**	0.2297	0.1310	0.1778
$\hat{P}$ **95% CI**	[0.1190, 0.3776]	[0.0695, 0.2443]	[0.0895, 0.3291]
$\hat{\phi}$ **mean**	0.7234	0.4926	0.5527
$\hat{\phi}$ **95% CI**	[0.2901, 0.9990]	[0.0654, 0.9048]	[0.1128, 0.9592]

Table 3.11: Results of MCMC simulations on experimental data sets for parameter estimates in the 'Estimating P and S simultaneously - S prior informed' case with AR(1) errors, detailing the priors, parameter estimates and 95% credibility intervals (CI) for estimates.

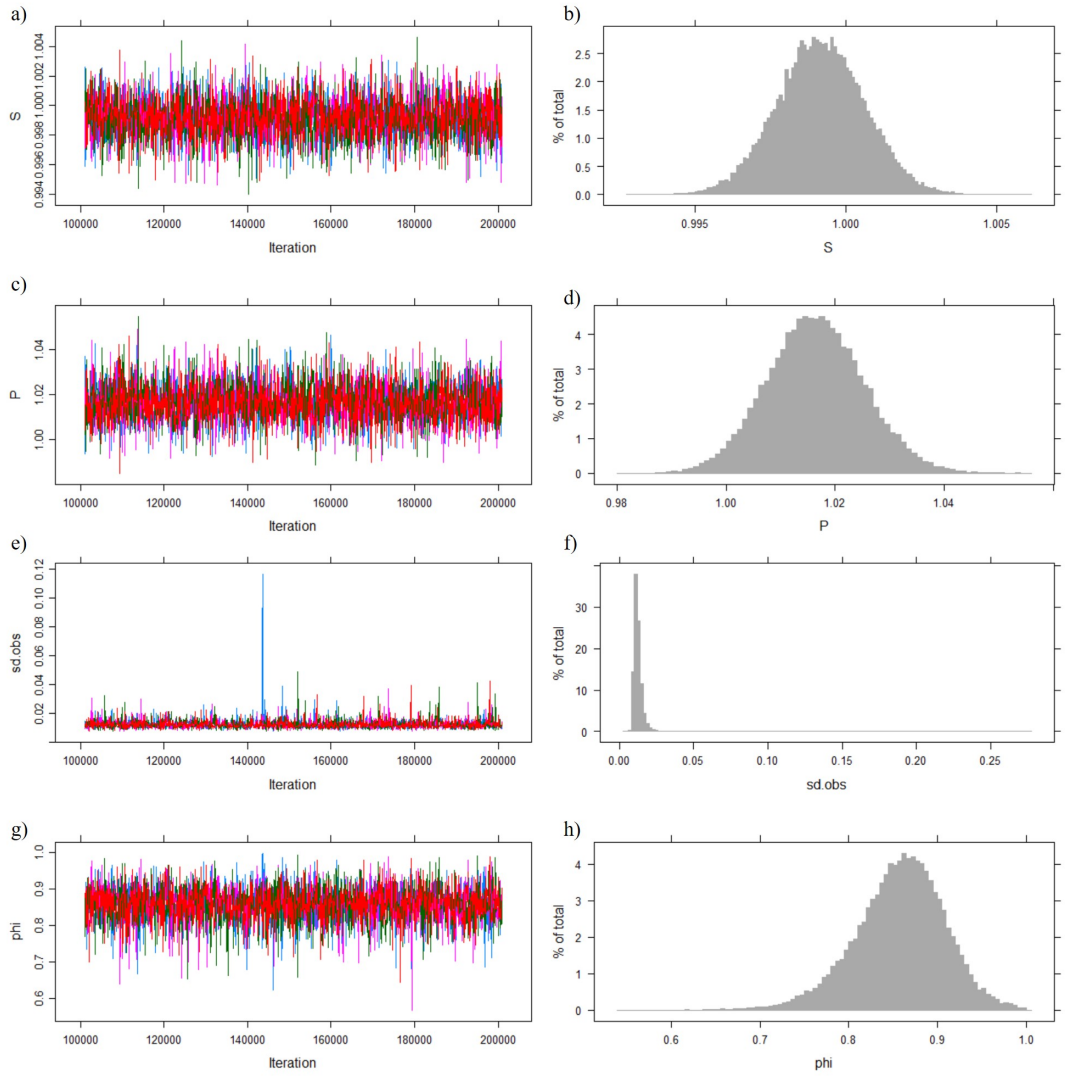

Figure 3.1: **Visual outputs from JAGS MCMC simulations for the 11001 *in silico* case where** $S = 1, P = 1, dt = 0.01$**.** Analysis is for the 'Estimating P and S simultaneously - S prior informed' case, with AR(1) errors assumed. Plots show traces and marginal posterior density histograms for each parameter monitored by the simulations, namely $\hat{S}$ **(a, b)**, $\hat{P}$ **(c, d)**, $\hat{sd.obs}$ **(e, f)** and $\hat{\phi}$ **(g, h)**.

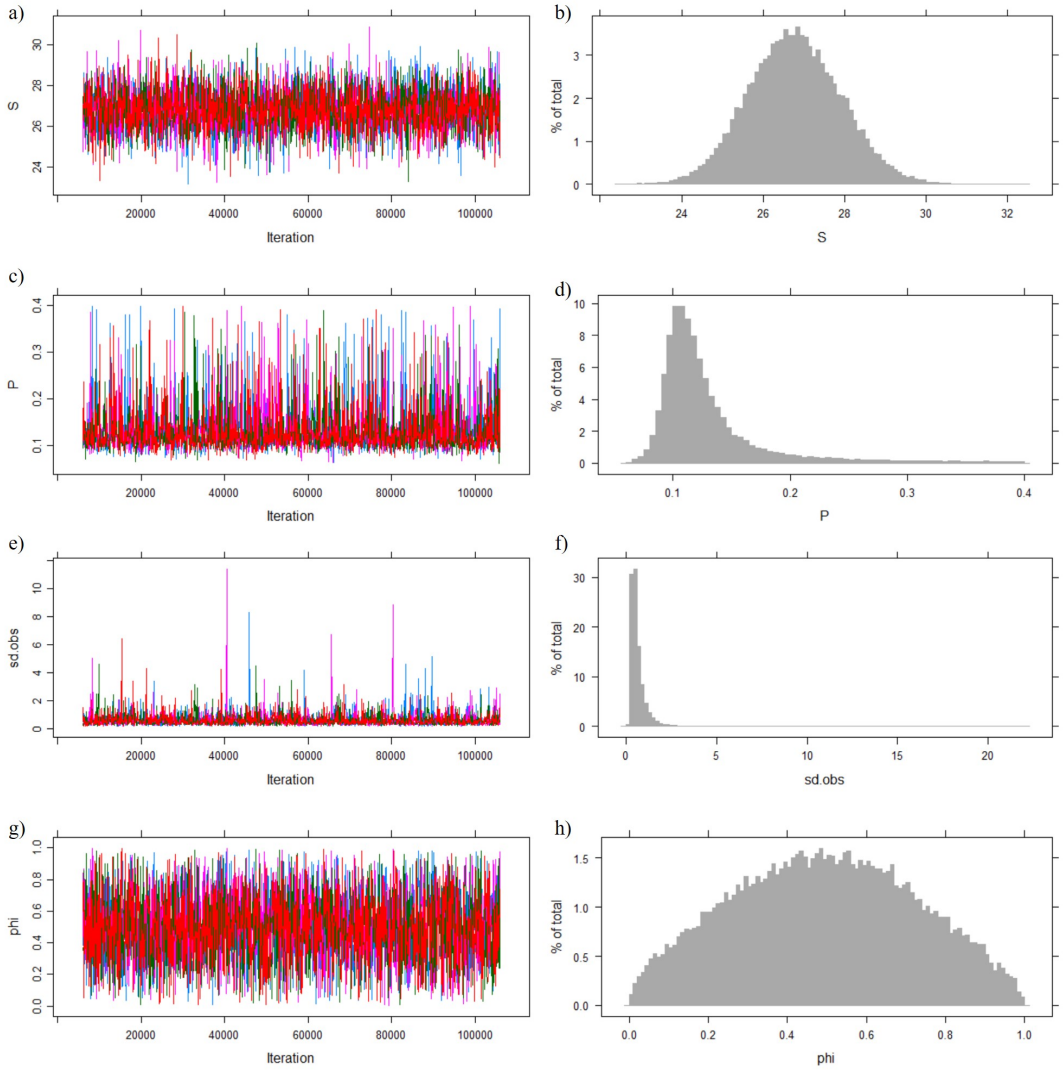

Figure 3.2: **Visual outputs from JAGS MCMC simulations for the Spheroid 2 experimental case.** Analysis is for the 'Estimating P and S simultaneously - S prior informed' case, with AR(1) errors assumed. Plots show traces and marginal posterior density histograms for each parameter monitored by the simulations, namely $\hat{S}$ **(a, b)**, $\hat{P}$ **(c, d)**, $\hat{sd.obs}$ **(e, f)** and $\hat{\phi}$ **(g, h)**.

observed in those analyses. Credibility intervals for both $\hat{S}$ and $\hat{P}$ are fairly wide, indicating that estimates are still uncertain. The credibility interval for the point estimate of $\hat{S}$ in Spheroid 3 is an exception to this being fairly narrow.

Correlation is again estimated as being different between data sets though all point estimates are reasonably high, ranging between posterior mean values of 0.4143 and 0.8576. The trace and marginal posterior density histograms for two of the simulations are given in figures 3.1 and 3.2 as an example, and it appears that all traces have covered the whole parameter space in each case and density histograms are unimodal.

Figure 3.3 shows priors and marginal posteriors for the experimental data sets plotted together to assess how much both the data and the prior have informed the marginal posterior distribution for each parameter of interest. We expect the prior density to be flatter than the posterior density if the analysis wasn't too strongly influenced by the prior.

The R package MCMCvis (Youngflesh, 2018) was used to create these plots, inputting priors by using 20,000 draws from the relevant distributions. The package calculates the percentage overlap between prior and marginal posterior densities using the inbuilt `overlap` function in R.

We see from figures 3.3a(i), b(i) and c(i) that the informative gamma priors used for S as estimated from the $RMSS$ data are very close to the marginal posterior for $\hat{S}$, as expected, meaning that both the $RMSS$ data and the $\ln(VACF)$ data are utilized to gain this estimate. Priors for S with twice the variance were tested, though this didn't make much difference to estimates and priors were thus left as being very informative, making use of all available data.

The priors for P are uniform in all cases over the interval [0.06, 0.4]. We see in figures 3.3a(ii), b(ii) and c(ii) that the prior densities are much flatter than the marginal posteriors for $\hat{P}$, which indicates that the posteriors for $\hat{P}$ were mainly affected by the data rather than the flat priors, as expected.

Finally, figures 3.3a(iii), b(iii) and c(iii) show the prior and marginal posterior densities for $\hat{\phi}$. The beta prior with parameters $a = b = 1.5$ was used in all cases and it seems that in the case of Spheroid 1, this prior was less informative than

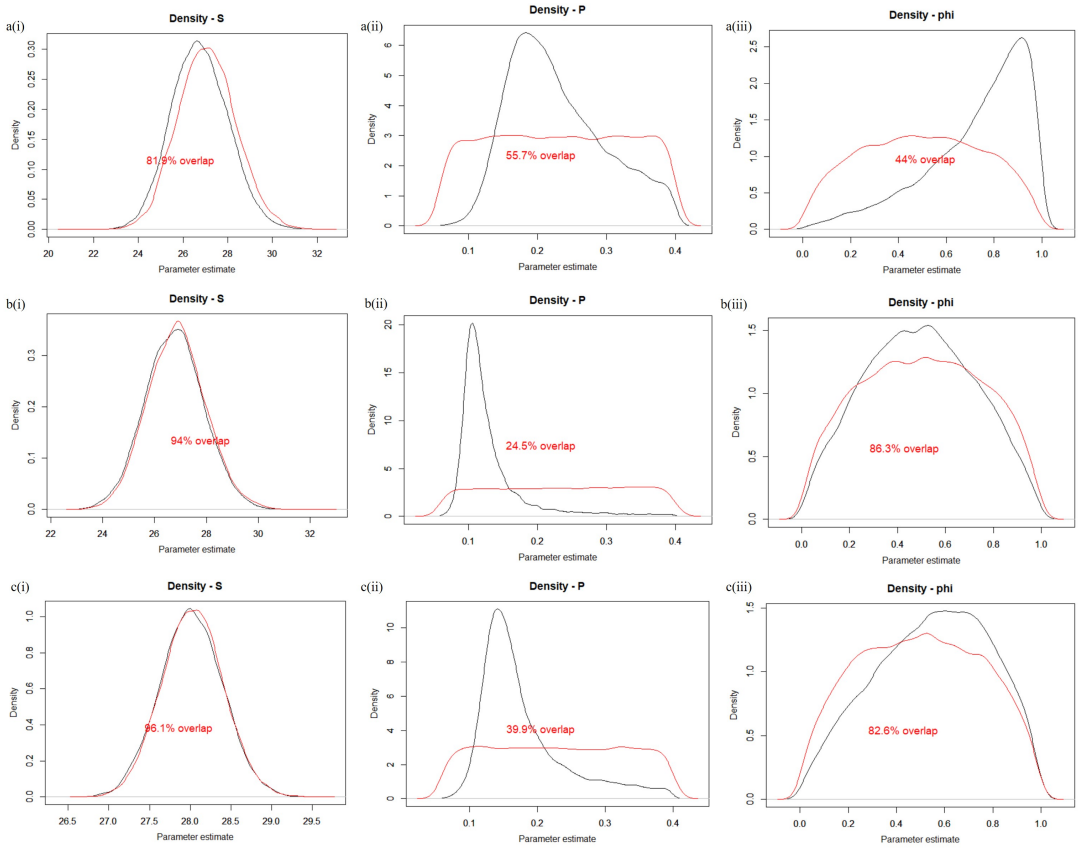

Figure 3.3: **Plots of prior and marginal posterior distributions for experimental data sets** in the 'Estimating P and S simultaneously - S prior informed' case, with AR(1) errors assumed. Plots show smooth estimates of prior densities formed from 20,000 draws from the relevant distribution (red) and the marginal posterior density (black) for each parameter of interest monitored by the simulations and a percentage overlap between the prior and marginal posterior densities as calculated by the R package MCMCvis (Youngflesh, 2018). For Spheroid 1 plots show marginal posterior densities as compared to priors **a)i** Gamma(433,16), **a)ii** Unif(0.06,0.4), **a)iii** Beta(1.5,1.5). For Spheroid 2 plots show marginal posterior densities as compared to priors **b)i** Gamma(564,21), **b)ii** Unif(0.06,0.4), **b)iii** Beta(1.5,1.5). For Spheroid 3 plots show marginal posterior densities as compared to priors **c)i** Gamma(5466,195), **c)ii** Unif(0.06,0.4), **c)iii** Beta(1.5,1.5).

for Spheroids 2 and 3. Though in all three cases the posterior is tighter than the prior, so the data has informed the posterior distribution somewhat.

3.3 Model selection

3.3.1 Outline of approach

Model Selection criteria

For the analyses completed here, three model selection methods will be utilized to determine which model describes a set of data best; the Deviance Information Criterion (DIC) (Spiegelhalter *et al.*, 2002), the Widely Applicable Information Criterion (WAIC) (Watanabe, 2010) and the Bayes Factor (BF) (Jeffreys, 1939; Good, 1979, 1985; Kass & Rafferty, 1995).

The first method considered is DIC, developed by Spiegelhalter *et al.* (2002). The DIC is said to be a Bayesian version of the Akaike Information criterion (Akaike, 1973), widely used for model selection in both frequentist and Bayesian applications, and is based on the principle of 'goodness-of-fit' + 'complexity' (Spiegelhalter *et al.*, 2014). DIC is calculated for data y and parameters θ as

$$DIC = -2\ln p(y|\hat{\theta}) + 2p_D$$

where $\hat{\theta} = E[\theta|y]$ is the posterior mean, $\ln p(y|\hat{\theta})$ is the log predictive density of the data given the posterior mean, and p_D is the effective number of parameters defined as

$$p_D = 2\left(\ln p(y|\hat{\theta}) - E[\ln p(y|\theta)]\right).$$

The expectation in the second term here is an average of θ over its posterior distribution. In practice, this quantity can be computed using the formula stated in Gelman *et al.* (2013), namely

$$\text{Computed } p_{DIC} = 2\left(\ln p(y|\hat{\theta}) - \frac{1}{K}\sum_{k=1}^{K}\ln p(y|\theta^k)\right), \tag{3.5}$$

for simulations $\theta^k, k = 1, ..., K$ from, for example, running K iterations of MCMC.

The effective number of parameters can be thought of as the number of unconstrained parameters that can be estimated in the model. Some parameters for example may be constrained by their prior, i.e. with the condition that the parameter value be positive, and so will be mostly determined by this information and not by the model. Others may be informed by the prior but also the data and be less constrained (Gelman *et al.*, 2013). A totally constrained parameter where all information comes from its prior would contribute nothing to the number of effective parameters and a parameter totally estimated by the model with no prior information would contribute 1 to p_D. Most parameters would likely contribute some intermediate value being influenced by both the data and the prior to some extent.

WAIC developed by (Watanabe, 2010) is said to be a more fully Bayesian criterion for model selection, and more widely applicable than other available criteria in terms of models it can be used in conjunction with. WAIC is calculated using the log pointwise predictive density (lppd) and the effective number of parameters, p_W, where

$$\text{lppd} = \sum_{i=1}^{n} \ln \int p(y_i|\theta)p(\theta)d\theta,$$

for posterior distribution $p(\theta)$ and observations $y_i, i = 1, \ldots, n$, and

$$p_W = \sum_{i=1}^{n} \text{Var}[\ln p(y_i|\theta)].$$

Thus WAIC is given on the deviance scale by

$$\begin{aligned} WAIC &= 2p_W - 2\text{lppd} \\ &= 2\sum_{i=1}^{n} \text{Var}[\ln p(y_i|\theta)] - 2\sum_{i=1}^{n} \ln \int p(y_i|\theta)p(\theta)d\theta, \end{aligned}$$

and for practicality this can be computed using expressions from Gelman *et al.* (2013) as

$$\text{Computed}_{WAIC} = 2\sum_{i=1}^{n} V_{k=1}^{K}(\ln p(y_i|\theta^k)) - 2\sum_{i=1}^{n} \ln \left(\frac{1}{K} \sum_{k=1}^{K} p(y_i|\theta^k) \right), \quad (3.6)$$

where $V_{k=1}^{K}(.)$ signifies taking the variance over the MCMC sample of size K.

It is noted that there are different ways to define the penalties used in both the DIC and WAIC, but in this work we only focus on those penalties defined here. In practice we take the DIC and p_W as calculated by the `dic` module in the rjags package (Plummer, 2019), and subsequently calculate the lppd to obtain the WAIC.

The final method of model selection considered here, the Bayes Factor, can be thought of as the Bayesian equivalent of hypothesis testing. Described by Jeffreys (1939); Good (1979, 1985) and Kass & Rafferty (1995), it is the ratio of the marginal likelihoods, given for data D and models under comparison M_1 and M_2 by

$$B_{12} = \frac{p(D|M_1)}{p(D|M_2)}, \tag{3.7}$$

giving the odds in favour of the 'null' model, M_1.

There can be difficulty calculating the Bayes Factor due to the use of marginal likelihoods, and so as nested models are considered in this work, the Savage-Dickey density ratio will be used (Dickey & Lientz, 1970). Here, it is assumed there are parameters ϕ allowed to vary in the alternative model M_2 but all other parameters ψ are the same as those in the null model M_1. The null model is then the same as the alternative model M_2 but with ϕ set to fixed values ϕ_0.

There are also the assumptions of equal likelihoods i.e.

$$P(D|M_1) = P(D|\phi = \phi_0, M_2),$$

and prior continuity i.e.

$$\lim_{\phi \to \phi_0} P(\psi|\phi, M_2) = P(\psi|M_1)$$

(Wagenmakers *et al.*, 2010).

Thus the ratio in 3.7 can be written as

$$B_{12} = \frac{p(D|M_1)}{p(D|M_2)} = \frac{P(\phi = \phi_0|D, M_2)}{P(\phi = \phi_0|M_2)}, \tag{3.8}$$

which becomes the ratio of the posterior distribution at the fixed parameter value(s) given the data and the alternative model, and the prior distribution at the fixed parameter value(s) given the alternative model.

Once calculated, the Bayes Factor is interpreted as odds in favour of the null model, so its value indicates the strength of evidence for the simpler model. Jeffreys interpretations of the Bayes Factor are used in this work, and these suggest that a Bayes Factor greater than 1 says that the null model is preferred, whereas a value less than 1 signifies that the alternative model is preferred. A full interpretation of the Bayes Factor can be found in Jarosz & Wiley (2014), showing some of the different perspectives that have been suggested over the years.

Application of model selection criterion to the analysis from the framework

Along with using a regression model with correlated errors assumed to follow an AR(1) process, an AR(2) process was considered to see if using a higher order correlation might better explain the experimental data. This was due to the frequentist analysis in the original framework not being able to explain the experimental data as well as the *in silico* data.

Thus, an alternative regression model incorporating an AR(2) process was considered, namely

$$\ln(VACF)_i = \ln(S^2) - \frac{1}{P}t_i + \epsilon_i, \tag{3.9}$$

for time step i and observations of $\ln(VACF)$ at the i^{th} time step, where $\epsilon_i = \phi_1\,\epsilon_{i-1} + \phi_2\,\epsilon_{i-2} + u_i$ and $u_i \sim N(0, 1/\tau)$. The AR(2) process is seen in the errors which are dependent on the errors in the previous two time steps, thus ϕ_1 is the first order correlation coefficient and ϕ_2 the second order correlation coefficient.

Upon conducting Bayesian analysis of the regression model with AR(2) errors it was clear that the marginal posterior distribution of the second order correlation coefficient ϕ_2 was not centred around 0 as would be expected if the AR(1) model was sufficient to describe the observed data. This is seen from the results in tables 3.12 and 3.13 and the visuals in figures 3.4j) and 3.5j).

Model selection was then carried out comparing the models with AR(1) and AR(2) errors, these models being nested, using the three criteria outlined above - DIC, WAIC and the Bayes Factor calculated by means of the Savage-Dickey

density ratio. Analyses are conducted as in the analytical case from section 3.2.5, that for which informative priors for S are used.

The Savage-Dickey density ratio was calculated using equation 3.8, setting $\phi_2 = 0$ in the AR(2) model and using the logspline package (Kooperberg, 2020) in R to obtain the value of the marginal posterior density for ϕ_2 at this point. It is noted that in the Bayes factor, the AR(1) model is considered M_1, or the null model and AR(2) is M_2, the alternative. All required assumptions are satisfied, with priors for ϕ_1 and ϕ_2 chosen such that their joint distribution in the AR(2) model tends to that of the ϕ_1 prior in the AR(1) model with $\phi_2 = 0$. This is automatically satisfied in this case, as long as the priors for ϕ_1 are the same in both models, because priors for ϕ_1 and ϕ_2 are independent. The likelihoods of both models are also equal when $\phi_2 = 0$.

JAGS model code for the AR(2) model can be found in Appendix B.4 and example MCMC simulation code with model selection in Appendix C.2.

Results

Firstly, the analyses of the AR(2) models are presented for comparison with those in the AR(1) case for the estimates in section 3.2.5. Priors for ϕ_2 were kept uniform over $[-1, 1]$, again since there was no prior knowledge of its value at this point. Values can be seen in tables 3.12 and 3.13 and visuals in figures 3.4 and 3.5. Again, the trace plots and marginal density histograms suggest that chains have converged and mixed well.

We note that point estimates of $\hat{S}$ and $\hat{P}$ are close to those obtained with the AR(1) model, though for the experimental cases we see negative second order correlation for Spheroids 1 and 3. The credibility intervals for $\hat{\phi_2}$ in these spheroids suggest that this negative correlation may not be significant as they are inclusive of 0. All credibility intervals for $\hat{\phi_1}$ confirm that positive correlation is observed at first order.

Results of the model selection criteria calculations are given in table 3.14. For *in silico* data sets, the Bayes Factor can be interpreted as saying there is decisive evidence in favour of AR(2) in the 11001 case and anecdotal evidence in

	11001	**2501**
Sample size	59	5
S **prior**	Gamma(77,78)	Gamma(100183, 4000)
P **prior**	Gamma(4,0.25)	Unif(0.06,0.4)
ϕ_1 **prior**	Beta(2,2)	Beta(1.5,1.5)
ϕ_2 **prior**	Unif(-1,1)	Unif(-1,1)
τ **prior**	Gamma(0.001, 0.001)	Gamma(0.001, 0.001)
Initial values (S, P, ϕ_1, ϕ_2, τ)	1, 1, 0.5, 0.5, 1	25, 0.1, 0.5, 0.5, 1
Chains	4	4
Iterations, burn-in	371,410, 100,000	100,000, 100,000
$\hat{S}$ **mean**	0.9995	25.0428
$\hat{S}$ **95% CI**	[0.9970, 1.0019]	[24.8865, 25.1952]
$\hat{P}$ **mean**	1.0139	0.1013
$\hat{P}$ **95% CI**	[0.9992, 1.0294]	[0.0868, 0.1148]
$\hat{\phi_1}$ **mean**	0.7761	0.3920
$\hat{\phi_1}$ **95% CI**	[0.6231, 0.9292]	[0.0184, 0.7881]
$\hat{\phi_2}$ **mean**	0.5991	0.3894
$\hat{\phi_2}$ **95% CI**	[0.3891, 0.7921]	[-0.5641, 0.9950]

Table 3.12: Results of MCMC simulations on *in silico* data for parameter estimates in the 'Estimating P and S simultaneously - S prior informed' case with AR(2) errors, detailing the priors, parameter estimates and 95% credibility intervals (CI) for estimates. 11001 refers to the data set where $S = 1$, $P = 1$ and 2501 to the data set where $S = 25$, $P = 0.1$.

	Spheroid 1	**Spheroid 2**	**Spheroid 3**
Sample size	5	6	5
S **prior**	Gamma(433,16)	Gamma(564,21)	Gamma(5466,195)
P **prior**	Unif(0.06,0.4)	Unif(0.06,0.4)	Unif(0.06,0.4)
ϕ_1 **prior**	Beta(1.5,1.5)	Beta(1.5,1.5)	Beta(1.5,1.5)
ϕ_2 **prior**	Unif(-1,1)	Unif(-1,1)	Unif(-1,1)
τ **prior**	Gamma(0.001, 0.001)	Gamma(0.001, 0.001)	Gamma(0.001, 0.001)
Initial values (S, P, ϕ_1, ϕ_2, τ)	27, 0.08, 0.5, 0.5, 1	27, 0.08, 0.5, 0.5, 1	27, 0.08, 0.5, 0.5, 1
Chains	4	4	4
Iterations, burn-in	100,000, 100,000	100,000, 100,000	100,000, 100,000
$\hat{S}$ **mean** (μ/**h**)	27.0047	26.7107	28.0274
$\hat{S}$ **95% CI**	[24.4861, 29.5136]	[24.5040, 28.8745]	[27.2720, 28.7520]
$\hat{P}$ **mean (h)**	0.1899	0.1153	0.1456
$\hat{P}$ **95% CI**	[0.1134, 0.3354]	[0.0715, 0.1826]	[0.0914, 0.2247]
$\hat{\phi_1}$ **mean**	0.3336	0.3430	0.3310
$\hat{\phi_1}$ **95% CI**	[0.0363, 0.6425]	[0.0154, 0.7188]	[0.0154, 0.6804]
$\hat{\phi_2}$ **mean**	-0.1123	0.0936	-0.0983
$\hat{\phi_2}$ **95% CI**	[-0.8004, 0.3887]	[-0.6128, 0.8059]	[-0.7494, 0.5856]

Table 3.13: Results of MCMC simulations on experimental data for parameter estimates in the 'Estimating P and S simultaneously - S prior informed' case with AR(2) errors, detailing the priors, parameter estimates and 95% credibility intervals (CI) for estimates.

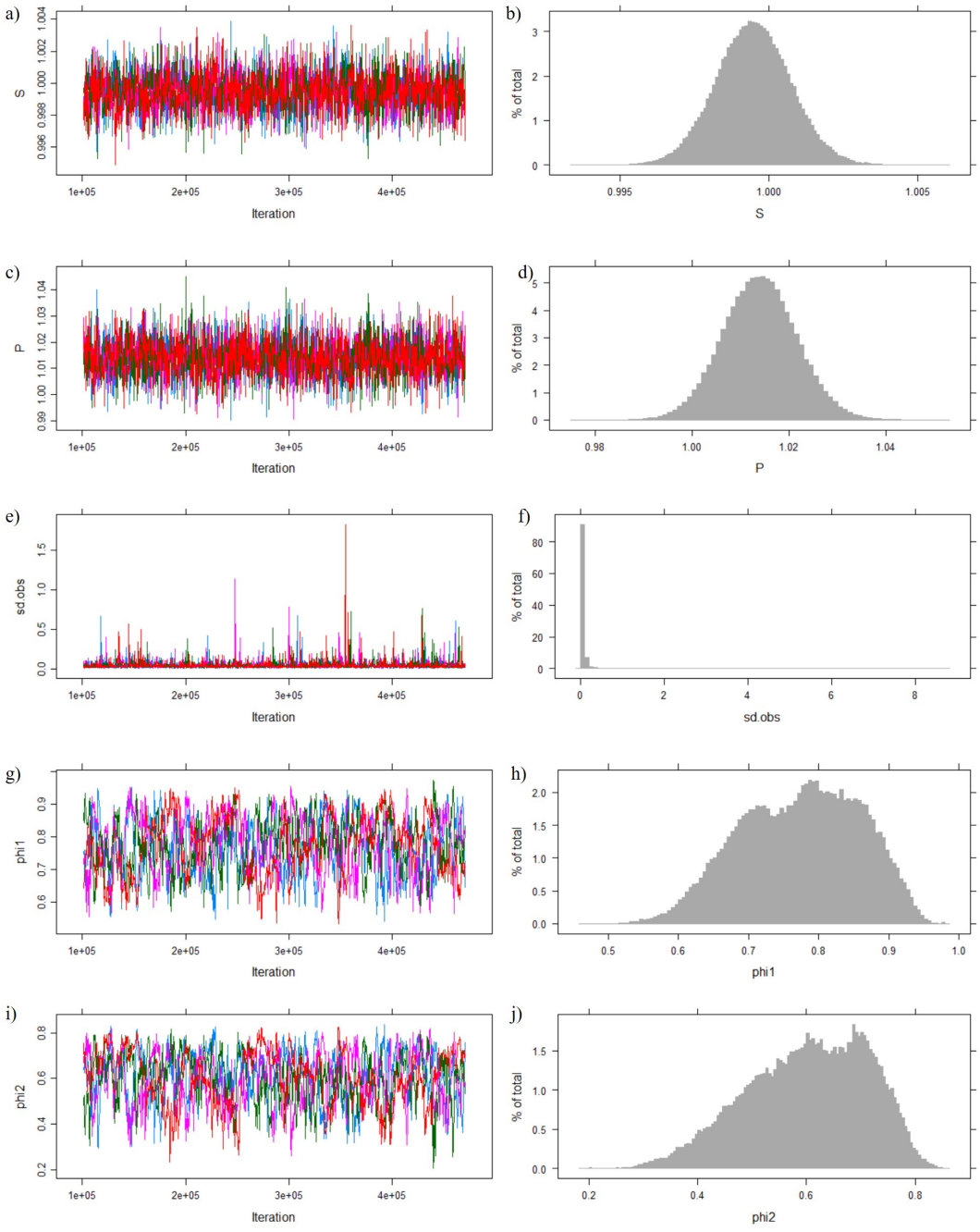

Figure 3.4: **Visual outputs from JAGS MCMC simulations for the 11001 *in silico* case.** Analysis is for the 'Estimating P and S simultaneously - S prior informed' case, with AR(2) errors assumed. Known parameter values are $S = 1, P = 1$. Plots show traces and marginal posterior density histograms for each parameter monitored by the simulations, namely $\hat{S}$ **(a, b)**, $\hat{P}$ **(c, d)**, $\hat{sd.obs}$ **(e, f)**, $\hat{\phi}_1$ **(g, h)** and $\hat{\phi}_2$ **(i, j)**.

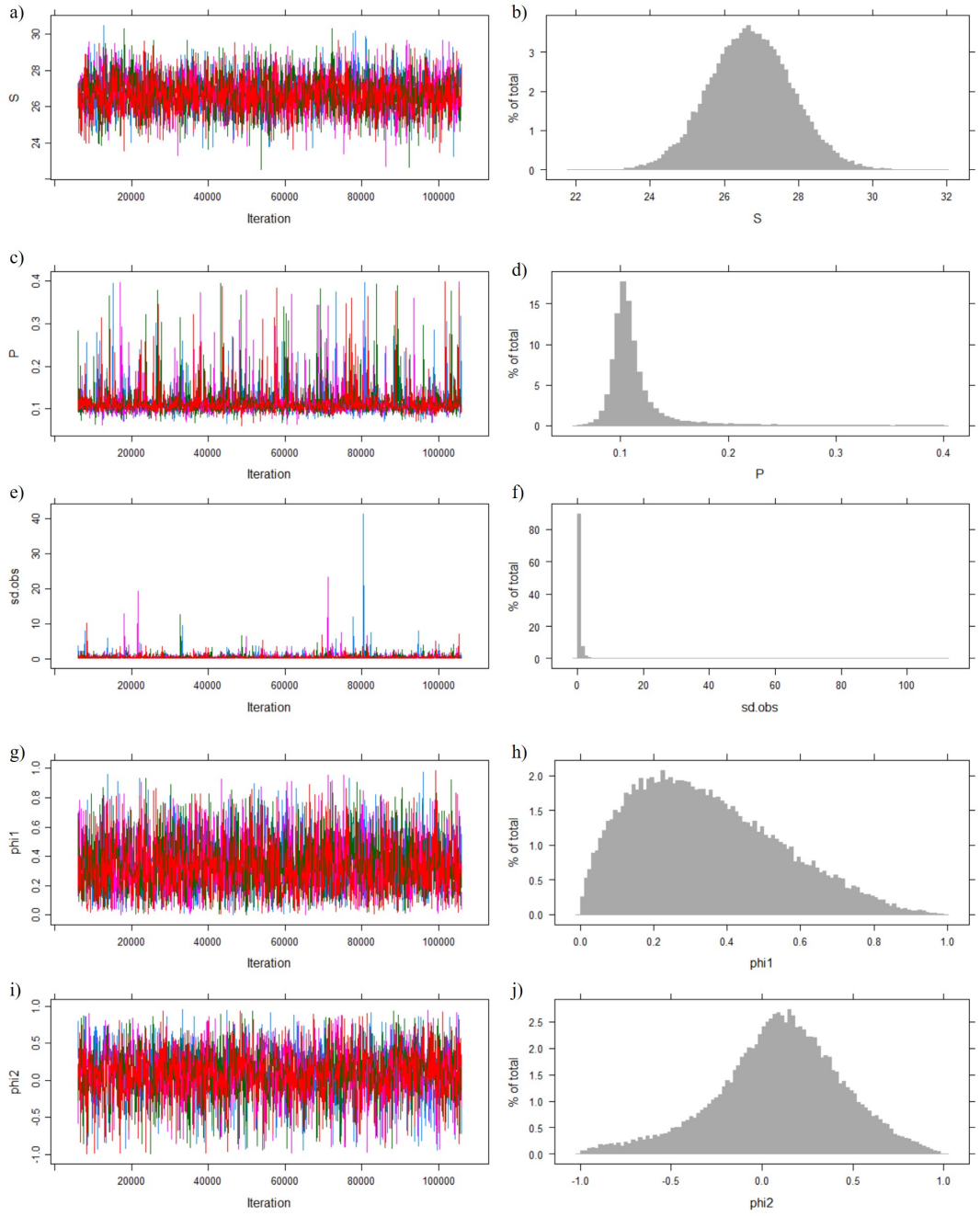

Figure 3.5: **Visual outputs from JAGS MCMC simulations for the Spheroid 2 experimental case.** Analysis is for the 'Estimating P and S simultaneously - S prior informed' case, with AR(2) errors assumed. Plots show traces and marginal posterior density histograms for each parameter monitored by the simulations, namely $\hat{S}$ **(a, b)**, $\hat{P}$ **(c, d)**, $\hat{sd.obs}$ **(e, f)**, $\hat{\phi}_1$ **(g, h)** and $\hat{\phi}_2$ **(i, j)**.

	DIC		WAIC		BF
Data set	**AR1**	**AR2**	**AR1**	**AR2**	**AR1 vs AR2**
11001	-438.3	-456.4	-388	-272	1.0380e-05
2501	-2.82	21.62	-11.4	-8.31	0.7829
Spheroid 1	6.81	51.63	11.3	15	4.3018
Spheroid 2	10.65	12.71	10.1	8.99	2.3877
Spheroid 3	6.89	46.33	6.45	7.65	2.6795

Table 3.14: Results of model selection criteria for all 5 data sets. For DIC and WAIC the model with the smaller value is preferred and is underlined. A Bayes Factor greater than 1 signifies that the AR(1) model is preferred, whereas a value less than 1 signifies that the AR(2) model is preferred.

favour of AR(2) in the 2501 case. When looking at experimental data sets the Bayes Factor says that there is moderate evidence in favour of AR(1) in the case of Spheroid 1, and anecdotal evidence in favour of AR(1) in the Spheroid 2 and Spheroid 3 cases.

This means that none of the criteria give a unanimous conclusion, though in most cases the AR(1) model is the preferred one. The AR(2) model is only preferred among the experimental data in the Spheroid 2 case when WAIC is calculated, but is preferred in both of the *in silico* cases when the Bayes Factor is calculated, and in the 11001 case when DIC is calculated.

Since the Bayes Factor gives a quantitative measure of the strength of evidence in favour of a certain model, we particularly note how strongly it suggests that the AR(2) model is preferred in the 11001 case. This could be due to the fact that there is a larger sample size in this data set, meaning that correlations can be monitored over a longer time period and estimated more accurately, but also that a simpler model would be preferred in data sets with very small sample sizes due to a more complicated model being less parsimonious. The AR(2) model is trying to fit 5 parameters and so this would be selected against in a data set with only 5 or 6 observations, so in all of the cases other than the 11001 data set. There would be a lot of uncertainty in these parameter estimates, leading

to a flat posterior. The evidence for the AR(1) model is however also not very strong according to the Bayes Factors for the other data sets, suggesting that larger sample sizes might give alternative results.

These results must however be taken with caution given the issues with sample size and also that the Bayes Factor relies on choice of prior and the quality of the estimation of the posterior distribution, here through MCMC. WAIC is said to be 'fully Bayesian' because it uses the posterior distribution, rather than conditioning on a point estimate (Gelman *et al.*, 2013) as is done in DIC, and it could be said that in this context it would be natural to trust the results of WAIC more than those from DIC. If the posterior distribution is not well summarized by its mean then DIC can give nonsensical results (Gelman *et al.*, 2013).

For comparison, an FGLS model with AR(2) errors was fitted to the $\ln(VACF)$ data from the 11001 data set in MATLAB within the frequentist framework to see how the estimates for S and P compared. The results were $\hat{S} = 0.9978,\ [0.9892, 1.0064]$ and $\hat{P} = 0.9843, [0.9220, 1.0556]$. These estimates are slightly worse than the Bayesian AR(2) estimates based on comparison with the known values. When comparing frequentist AR(1) to AR(2), the estimates for S are identical, as expected, but the AR(2) P estimate has a wider confidence interval, with the point estimate being very similar.

3.4 Bayesian Analysis: What's better?

It is easily seen that there are many advantages to the Bayesian approach for the analysis of data, and more specifically tracking data. In the wider sense, it allows a picture of uncertainty to be built up in the parameter estimates rather than just outputting a point estimate along with its confidence interval. This shows the most likely value of a parameter but still demonstrates how uncertain these values are. It must be appreciated that model outputs have an inherent degree of variation and this in turn affects the reliability and validity of any predictions made. Running simulations for long enough however can ensure that any variability is negligible. Bayesian analysis is also capable of handling more complex problems than frequentist analysis, and things like multiple comparisons

are handled automatically. The phenomenon known as 'Ockham's Razor' also applies, meaning that Bayesian analysis favours simpler models which sufficiently explain the data (Berger, 2006).

In terms of interpreting this uncertainty, another advantage of the Bayesian approach is the credibility interval that results from the estimation. By definition the true parameter value falls within the $100\alpha\%$ interval with probability α. This is a simple interpretation and gives a quantitative idea of the uncertainty, in contrast to the classical confidence interval, which is often misinterpreted and mistaken for having the same definition as the credibility interval. This extends to other concepts in statistics that are often misunderstood in the frequentist approach, like the p-value, and as such Bayesian analysis provides the advantage that conclusions from analyses are often more easily interpreted.

Specifically in relation to the data and problem being considered in this work, there are various advantages of the Bayesian approach over the frequentist. The ability to incorporate prior information and data is one of the main advantages of the approach, and even when there is little prior information this can be represented by an uninformative prior.

Estimates of P will be quite uncertain since there is not much in the way of prior knowledge about the true value of this parameter, at least for GBM cells. This may also be the case for other cell types, and even if information on persistence time of cell types could be obtained, this may be costly to obtain, both in time and money, and so we can assume it is largely not known. This being the case, we are at an advantage using the Bayesian approach here and being able to include less informative priors for P.

We are also able to take correlation into account directly which we know is inherent in the data. This is done by directly imposing the AR(1) process onto the observations or the errors as necessary and allows freedom of choice in how that correlation might be modelled by its prior.

We are also able to easily incorporate both the $RMSS$ and $\ln(VACF)$ data here to gain better estimates of parameters of interest. Using informed priors for S which have been derived from the S estimates obtained using the $RMSS$

data has allowed implicit use of this time series data when implicitly estimating S from the $\ln(VACF)$ data. This in turn should improve the accuracy of the P estimates as the intercept of the regression line used should also be more realistic.

Finally, we could also here make use of RJMCMC, as outlined in the introduction to this chapter, and use the power of Bayesian analysis to select the most appropriate AR process to include in the model. It should be noted however that the small sample sizes in the data sets here are limiting factors to this as it is desirable to avoid overfitting.

3.5 Bayesian Analysis: What's worse?

Just as it is an advantage of the Bayesian approach, prior knowledge could also be seen as a hindrance in these analyses. Given that we don't have much knowledge about the true values of P in experimental cases and the prior has to thus be flat, we would ideally need a large data set to allow accurate estimation of the posterior density. Because in the experimental cases we also have very small sample sizes in the $\ln(VACF)$ data, when it comes to estimating P we are stuck with little data as well as the flat prior. This is a problem in the frequentist analyses also, and the Bayesian and frequentist estimates are quite different. In *in silico* cases, point estimates of $\hat{P}$ are consistent with the known values and have small credibility intervals, particularly when the priors for S are informative in the simultaneous estimations.

Bayesian analysis is also widely criticised for its subjectivity, mainly attributed to the fact that priors are chosen by the analyst. In a good Bayesian analyses these priors will be carefully considered and justified, and experts should be consulted to gather appropriate information. It is also possible to carry out objective Bayesian analysis and Berger (2006) makes some good arguments for this type of analysis as well as addressing criticisms on the subjective nature of Bayesian statistics.

One other drawback to the Bayesian approach which is specific to the use of MCMC simulations is that we are only ever working with an approximation to the posterior and never the exact distribution. This means that convergence

needs to be checked and chains don't always mix well, meaning that analysis can be more unreliable than in the frequentist approach. The impact of this can be reduced by careful monitoring of convergence and running simulation for longer times.

3.6 Discussion and Conclusions

This chapter has presented statistical analyses of 5 data sets as used in chapter 2, with the intention of replicating analyses conducted using the frequentist framework for modelling 3-dimensional cell tracking data with the Persistent Random Walk model. MCMC simulations were carried out using JAGS to obtain estimates of parameters of interest for the model and surrounding uncertainty for comparison with the frequentist equivalents obtained previously.

Four different analytical cases were considered for obtaining the best possible estimates. Firstly cell speed parameter S was estimated using time series data from the calculation of $RMSS$ and then the persistence time P was estimated using a regression model fit to the $\ln(VACF)$ data with a generic intercept. Moving on to estimate S and P simultaneously, the cases where priors are vague and then informed using the estimates of S based on the $RMSS$ were considered.

The point estimates obtained were generally comparable to the frequentist estimates, with those from the *in silico* data sets being close to the known values. The experimental cases were consistent with the frequentist estimates too for the S parameter, though $\hat{P}$ point estimates were markedly higher than those obtained in the frequentist analyses. Confidence intervals and credibility intervals are however wide in both cases, more so in the Bayesian analyses of experimental data sets, thus there is still a large degree of uncertainty in these estimates.

Model selection was also carried out comparing the fit of the model with assumed AR(1) errors to that with AR(2) errors. To do this, the DIC, WAIC, and Bayes Factor were calculated for all models and the results showed that overall the model with AR(1) errors was preferred in most cases according to all 3 criteria. This is however not the case for the 11001 *in silico* data set according to the DIC and Bayes Factor which say it is better described when assuming

AR(2) errors, with the Bayes Factor presenting decisive evidence that this is the case. This could be an indicator that the other data sets simply need larger sample sizes to reach this conclusion, as with only 5 or 6 observations it is highly unlikely that a model as complex as that involving an AR(2) process would be chosen as the better fit. We see that even in the 2501 case the Bayes Factor gives anecdotal evidence in favour of the AR(2) model, perhaps supporting this hypothesis further.

Overall the Bayesian approach to this work has uncovered some interesting results about the framework and the data being used. It reiterates the conclusion from chapter 2, that data needs to be collected in shorter time steps in order to provide bigger sample sizes before the cut-off point. This will allow further investigation of the conclusions made here.

Further studying the correlation present in the data, whilst not specific to the Bayesian method, has revealed that correlation of higher order than that modelled by an AR(1) correlation structure is present, something which was not previously considered but which came to light when looking at model selection. It would be interesting to look into this further, testing higher orders of correlation even, though more data and larger sample sizes are needed to facilitate this. RJMCMC could be used on larger data sets to study the appropriate degree of correlation to include in the model, and this method would allow the jumping between parameter spaces necessary for testing several different AR models.

An advantage of using Bayesian analysis here has been the ability to include prior information in the model. Given the small sample sizes, prior information will carry more weight in the resulting marginal posteriors and here several different types of prior have been considered. The choice of prior for S improved estimates greatly when an informed prior was used, this being a reliable method because S is estimated first, independently of P, using the $RMSS$ data. This also allowed the use of two different data sources when estimating the parameters, something that wasn't possible in the frequentist analysis. The Bayesian parameter estimates are also close to the true values where *in silico* data is used. When P is later estimated using these informed priors, the Bayesian analysis

gives higher values than the frequentist analysis, despite using flat priors for P in most of these analyses, further perpetuating the mystery of the true value(s) of P for the experimental data sets.

As is common with experimental data sets, we can assume that there is some intrinsic noise present in this data. This is largely accounted for by the stochastic nature of the model, but there may also be noise from measurement error in velocities and positions, which may affect results obtained from the MCMC simulations. An extra term could be added to the regression models to allow for variability in these measurements, as additive noise, for example a Gaussian term with constant variance. We would expect that any noise in the data would impact the results to a greater extent when the $\ln(VACF)$ data is used due to the sample sizes there being very small, though when $RMSS$ data is used, this noise should have less impact on the estimates of S.

To further the work in this chapter, the possibility of using alternative Bayesian methods could be considered, for example creating an ABC scheme to estimate parameters or looking further at validating and calibrating the analyses as in Hawkins-Daarud *et al.* (2013). Although we here assume an AR process describes the errors present when modelling both the $RMSS$ and $\ln(VACF)$, we are unsure if this is the most appropriate description. It may also be beneficial to develop this technique in the case that further models for these data are more complex or incorporate higher degrees of correlation. ABC could also be applied directly to the PRW SDE, using summary statistics like the $VACF$, MSD and $RMSS$ to obtain parameter estimates. This way goodness-of-fit could again be studied, and even improved upon using this method.

As with the work in previous chapters, more data is needed to look further into estimates of parameters from the experimental data sets. This would allow a clearer picture of the parameter space to be built up and the movement of these cells further studied.

This chapter however has provided a first look at comparing the frequentist and Bayesian methods of analysis as applied to the framework created for modelling 3-dimensional cell tracking using the PRW model. It has been shown that

parameter estimates in *in silico* cases are good but that experimental estimates are reasonably uncertain, though there are clear advantages to the Bayesian methodology that suggest many avenues for further investigation.

Chapter 4

Modelling chemotaxis in surface-attached bacteria

Bacteria are abundant in most environments that are interesting to study, making it essential that we understand the way they work if we are to gain a full understanding of such environments. An aspect that should be given particular consideration is chemotaxis, the processes by which bacteria move toward or away from chemicals. This phenomenon occurs in swimming and surface-attached bacteria, and can take on different forms in different strains. In this chapter we take a closer look at chemotaxis in surface-attached bacteria moving in 2 dimensions and explore the purpose of so called 'twiddles' which have been newly observed as part of chemotaxis in *Pseudomonas aeruginosa*.

4.1 Introduction and literature review

P. aeruginosa is one of the most prolific pathogens worldwide, causing a range of infections and numerous deaths year upon year and is one of the key enemies in the fight against antimicrobial resistance (Strateva & Yordanov, 2009; Bassetti *et al.*, 2018). For this reason, studying the behaviour of *P. aeruginosa* may shine new light onto how these cells move and survive in the environments they inhabit, allowing new strategies for infection prevention, treatment and control to be developed. We here attempt to add to the pool of knowledge by studying

chemotaxis in surface-attached *P. aeruginosa* and trying to understand how it can be modelled mathematically.

Chemotaxis has been studied extensively from a mathematical modelling point of view since the 1970s. Most models proposed deal with swimming bacteria, with a particular focus on the model organism *Escherichia coli*. The mechanisms by which *E. coli* carry out chemotaxis are well known and their pattern of motion is seen to be run-and-tumble. This is where the bacterium travels forward in a mostly straight line - the run - and then upon at least one of the flagellum reversing its direction of rotation and the flagellar bundle becoming uncoiled, tumbles to a new random orientation before moving off in the new direction with the bundle back to being coiled as normal (Lauga, 2016).

Arguably the most replicated mathematical model of chemotaxis was proposed by Keller & Segel (1971) after they questioned what happens to bacteria in a gradient with the aim of modelling experimental findings by Adler (1966, 1969). These experiments saw bacteria aggregating in chemotactic bands, proving that bacteria were actively moving up the attractant gradient. The resulting Keller-Segel model of chemotaxis considers the population density as a whole and gives a system of macroscopic flux equations describing how the cell density and chemoattractant signal concentration change over time. Countless models based upon that of Keller and Segel have been proposed since, many of which are reviewed by Arumugam & Tyagi (2021) and Hillen & Painter (2009).

The pioneering work of Berg and Brown, (e.g. Berg & Brown, 1972; Brown & Berg, 1974) on 3-dimensional tracking of *E. coli* investigated the behaviour of *E. coli* in an isotropic environment compared to that of a spatial gradient. The first of these works (Berg & Brown, 1972) describes the exponential distribution in the so-called 'twiddle' or tumble lengths of the cells, and that the probability of twiddling per unit time when moving in an isotropic environment is constant and can thus be modelled as a Poisson process. When travelling in a gradient however they find that the twiddling of bacteria moving down this gradient can be modelled as a Poisson process, but that cells moving up the gradient twiddle less, suggesting a non-constant turn rate. Further investigating this in their 1974

work (Brown & Berg, 1974), they find that positive gradients of chemoattractant suppress directional change whilst negative gradients have little effect. They conclude that the magnitude of response to the gradient formally depends on the time rate of change of fractional amount of chemoreceptor bound. For example, a cell swimming up a gradient will find more attractant, more of it will bind to chemoreceptors on the cell and the cell will run for longer, suppressing twiddles.

Alt (1980) introduced the idea of modelling chemotaxis as a biased random walk on the individual level, using a stochastic process for the position of each individual and incorporating turn angle and mean speed. This followed on from the pioneering work of Patlak (1953) and considers particles moving according to a random walk with persistence and external bias, now known as biased correlated random walks. Correlated random walks happen when there is a degree of correlation, or persistence, between each successive step the walker takes. The next step is dependent only on the current one and the correlation between steps diminishes over time. The biased random walk describes a sequence of motion which has an overall consistent bias in a preferred direction (Codling *et al.*, 2008). Random walks where both of these phenomena are observed are termed biased correlated random walks. Stochastic modelling of chemotaxis along with biased and/ or correlated random walks allow cells to be modelled individually on a microscopic scale, as opposed to the continuum models proposed on a macroscopic scale for the entire population.

The related notions of velocity-jump and space-jump processes were introduced by Othmer *et al.* (1988) expanding on the work by Alt (1980) and Patlak (1953). The space-jump process describes the motion of an individual via a sequence of jumps in space after a stochastically defined waiting time, though this process is poor at capturing persistent motion or correlation in positions. The velocity-jump process looks similarly at jumps, but in velocity where the individual travels with a constant speed and an orientation sampled from some distribution, for some finite length of time. It then undergoes a stochastic reorientation event, the motion as a whole being modelled as a Poisson process with a constant tumble rate (Taylor-King *et al.*, 2015). The velocity-jump process is

a popular way to model chemotaxis and can be found in many studies relating to bacterial motion (e.g. Erban & Othmer, 2004; Rousset & Samaey, 2013; Erban & Othmer, 2005, 2007; Treloar *et al.*, 2011; Plaza, 2019; Harrison & Baker, 2018; Calvez *et al.*, 2015).

Chemotaxis models also often take into account behaviour at different scales, as already touched upon above. Continuum models look at variables on the macroscopic scale and are concerned with population density, diffusion and metrics like speed and turn angle distributions. Models like those of Keller & Segel (1971) and Othmer *et al.* (1988) often provide a comprehensive analytical description of a bacterial population, taking a top-down view of modelling, and are able to replicate what has been observed in experiments.

These types of models however are incapable of incorporating heterogeneity between individuals that may affect the population as a whole, for example a mutation in one cell that spreads as the population grows. Individual-based models (IBMs) or agent-based models (ABMs) look at the other extreme and consider each individual on the microscopic scale. This bottom-up approach sees stochastic processes governing the movement of individual cells, usually with a focus on position, velocity and turn angle as parameters. Models also exist on the so-called mesoscopic scale which tries to marry the two former approaches together and works on a mid-level between microscopic and macroscopic, as explained in Othmer & Xue (2013).

IBMs are becoming more widely used as a way to model biological phenomena, particularly in the fields of ecology and microbiology (Grimm, 1999; Ferrer *et al.*, 2008), as seen in the comprehensive reviews given by Hellweger & Bucci (2009) and Kreft *et al.* (2017). Many studies have made use of these models on the single cell level when considering chemotactic motion of bacteria. The development of IBMs for studying bacterial growth and movement has seen packages like BacSim (Kreft *et al.*, 1998, 2001), INDISIM (Ginovart *et al.*, 2002) and iDynoMiCS (Lardon *et al.*, 2011) being able to test out motility hypotheses around aggregation and the formation of biofilms along with the effects of nutrient availability and gradients of such nutrients on *in silico* bacterial populations.

The rationale for using IBMs to study the motion of cells and other organisms is that they offer the chance to create simulations of real-world phenomena without the need for explicit mathematical equations that describe whole populations, rather just expressions for the movement characteristics of individuals. They also allow for the possibility that there are differences between individuals, meaning that many population types can be modelled (Hellweger *et al.*, 2016).

The drawbacks of this approach however are that it becomes computationally expensive for increasing numbers of cells involved in the simulations, and that the models themselves are quite often difficult to understand and replicate. In order to try and overcome the latter of these issues, Grimm *et al.* (2006) developed a standard protocol for describing IBMs and ABMs. This suggests a standardised description of an IBM, giving space for an overview and purpose of the model, design concepts and a more detailed explanation of what the model contains and does.

Despite all of the types of models outlined above, the vast majority of the literature focuses on modelling swimming bacteria with a run-and-tumble mechanism of chemotaxis as in the case of *E. coli*. Work done on other bacterial species has seen the discovery of alternative patterns of motion during chemotaxis, those broadly being the run-stop and run-reverse-flick types of motion.

The run-stop motion observed in *Rhodobacter sphaeroides* sees the cells undergo runs and reorientation events similar to *E. coli* (Armitage *et al.*, 1999), however instead of actively tumbling in the same way as *E. coli* by rotating their flagellar bundle, it was thought that *R. sphaeroides* stop their flagellar motor and allow rotational diffusion to reorientate them before moving off in the new direction. Rosser *et al.* (2013) have modelled this type of chemotactic motion with a two-state hidden Markov model. More recently they have used a description of bacteria as self-propelled particles, governed by a Langevin stochastic differential equation, suggesting that *R. sphaeroides* actually undergo active reorientation through showing that these cells need more than Brownian motion to reorientate (Rosser *et al.*, 2014).

Vibrio alginolyticus were also found to exhibit a motility pattern different

to that of *E. coli* during chemotaxis, opting for a run-reverse-flick 3 step pattern (Xie *et al.*, 2010). This involves a run, a reversal where the cell switches direction and moves back on itself, and then an instantaneous flick into a new run at some angle, observed as being most likely 90 degrees from the direction of the alignment of the cell in the preceding run and reversal. Altindal *et al.* (2011) characterise this motility by defining the drift and diffusion coefficients relevant to the motion, as is common for descriptions of chemotaxis, and compare these along with mean displacement of cells and migration speed in a chemical gradient to that of *E. coli*. They suggest that the advantage of the 3-step response over a simple run-and-tumble motion is in the benefit to the cell of being able to backtrack to an earlier position if they find themselves moving down a concentration gradient unwillingly.

More recently Alirezaeizanjani *et al.* (2020) have looked at the three run modes of *Pseudomonas Putida*, uncovered by Hintsche *et al.* (2017). These bacteria can move in push, pull or wrapped modes, meaning that the flagellar bundle works in different ways to propel the bacterium through a fluid. Alirezaeizanjani *et al.* (2020) found that the pull mode could be largely neglected due to its scarcity, whilst in the wrapped mode there was a clear bias in run time meaning that longer runs were observed on average when cells moved up the gradient. The runs conducted in push mode were found to be unaffected by gradients in chemoattractant.

These varied patterns of motility demonstrate the need for alternative chemotaxis models to be developed, particularly as assumptions made by older models based on the motion of *E. coli* do not hold. If for example, as in the case of *V. alginolyticus*, a cell has two turn angles during one cycle of its migration pattern, then this cannot be modelled using a single Poisson process as the tumble rate in this instance will not be constant and successive steps in the cycle will depend on those that came before.

Yang *et al.* (2015) employ a non-Poissonian regulation scheme for the flagellar motor switch and use this to study the run-reverse-flick migration pattern. They suggest that this strategy allows a cell like *V. alginolyticus* to increase its search

radius in the forward step (the run) by creating a peak in the time-dependent diffusivity but then backtrack if necessary, greatly reducing its net displacement in one cycle if motion was down the chemical gradient.

There is significantly less known about chemotaxis in surface-attached bacteria than in swimmers, and it was only recently confirmed by Oliveira *et al.* (2016) that surface-attached bacteria do chemotax and are capable of sensing chemical gradients with submicron precision. This work suggests that *Pseudomonas aeruginosa* behave in yet another different way when carrying out chemotaxis, and later Wheeler (2020) characterised a new-found behavioural mechanism for their motion, referred to as 'twiddling'.

This twiddling motility was observed to occur in combination with the more well known reversals. A bacterium undergoing a twiddle will slowly turn in a circular motion, vaguely on the spot though subject to rotational diffusion, and using its pili to slowly pull itself around. The cell does this for some length of time and then exits the twiddle in some direction on the unit circle, likely different to the 180 degree reorientation in a reversal. Twiddles take minutes or even hours to complete and are much slower than any reorientation events previously reported. During a reversal there is translocation of the pili so that the head of the bacteria switches to the new direction after pili are relocated to the opposite pole of the cell. In a twiddle there is translocation such that the pili localise at both ends of the cell.

Oliveira *et al.* (2016) also found that cells reversed direction more frequently when moving away from chemoattractant and that they travelled 25% faster when moving up a concentration gradient than when not. They call this a 'pessimistic' strategy for chemotaxis - cells increase their tumble rate when moving down a gradient but otherwise tumble rate is basal. 'Optimistic' and 'bi-bias' strategies for chemotaxis are also defined in Bearon & Durham (2019). The optimistic strategy for chemotaxis is where cells suppress their tumble rate if they are moving up a gradient but otherwise tumble rate is basal. The bi-bias strategy is when cells both suppress their tumble rate when moving up the gradient and increase their tumble rate when moving down the gradient.

Following the discovery of the twiddle mechanism for chemotaxis, there were questions raised about the purpose of such movements. It is suspected that the slow twitching allows a bacterial cell to slowly sample different directions around the circle that it may take after the reorientation, biased by concentration gradients in the environment. The purpose of the work in this chapter is to explore some of these questions from a mathematical perspective and make some headway to providing answers. It is investigated through statistical analyses whether bacteria bias their exits from twiddles in a preferential direction up a concentration gradient. An initial IBM framework is then developed to look closer at how twiddles and reversals may differ.

Data from experiments on *P. aeruginosa* undertaken by Wheeler (2020) are analysed to assist in further characterising this newly observed mechanism and testing hypotheses about how bacteria may use it to aid efficient chemotaxis. The model for chemotactic strategies proposed by Bearon & Durham (2019) is studied in relation to this data and it is shown that more data is needed to further model the twiddling mechanism and study its effect on chemotaxis.

4.2 Methods

It is thought that twiddles may benefit surface-attached bacteria undergoing chemotaxis more than reversals alone as they allow slow scanning of the environment locally and facilitate a choice of direction instead of relying on random movement and migration patterns, for example, to reach food. This is thought to be particularly beneficial for these surface-attached bacteria, for example, in a biofilm where food is likely to be in a thin layer rather than for bacteria in a flow, where being moved around and sampling a large area is likely to aid finding food that can be consistently moving around in the environment. Consequently, it would be interesting to study what effect twiddles have on chemotactic drift in a bacterial population, both independently and when coupled with reversals, and see what this means for bacteria crawling on a surface.

In order to investigate this idea mathematically, it was decided that an IBM should be built around the tracking data collected from the *P. aeruginosa* ex-

periments where twiddles and reversals were observed. Automated data was collected from the cell tracks along with a small subset of manually analysed data in order to facilitate investigation of some necessary hypotheses prior to the building of an IBM. These hypotheses were important *a priori* for checking assumptions about the data and were instructive in choosing how to build the subsequent model. The automated tracking data set would then ideally be used in the IBM to draw conclusions about reversals and twiddles in a scenario where data is adequate for the model purposes, i.e. paired twiddle entries and exits, full information about numbers of cells and numbers that undergo reorientation. It is no surprise that orientation of cells is of great importance here and so we are mostly concerned with direction of entry to, or exit from, a twiddle, as explained below.

4.2.1 Experiments and Data

Throughout this work tumbles will be referred to, along with reversals and twiddles. Tumbles are a catch-all term for any reorientation event. Thus reversals and twiddles are different types of tumble. Whenever a 'twiddle' is mentioned, it will always refer to the novel twitching mechanism in Wheeler (2020). A schematic of the different types of cell movement is shown in figure 4.1 for clarity.

A reorientation event, or tumble, is considered a 'twiddle' if it satisfies the following conditions, taken from Wheeler (2020):

- Cells rotate slowly, taking longer than 10 minutes and typically up to 60 minutes, for a full rotation of 2π radians
- Cells remain attached to the surface at both cell poles for the duration of a twiddle, but neither cell pole is fixed in position
- Cells rotate in a single direction (either clockwise or counter-clockwise) for the entirety of a twiddle
- The average cell speed during the rotation period is greater than $0.08\,\mu$m/min

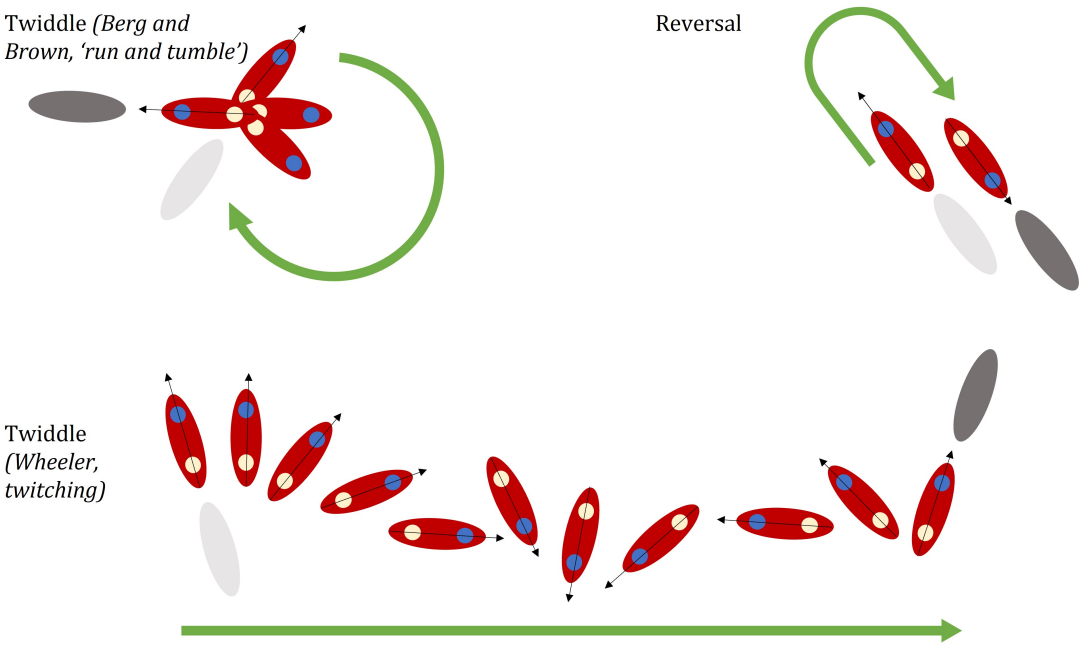

Figure 4.1: **Schematic of tumbles considered.** The figure depicts a 'twiddle' as per Berg & Brown (1972), commonly described as part of the 'run-and-tumble' mechanism typically observed in *E. coli*; a 'reversal' which is an instantaneous switch of the cell's leading pole and thus the direction of motion by half a turn, and a 'twiddle' as per Wheeler (2020), the novel twitching mechanism where cells move much slower and the leading pole remains the same throughout. Blue circles indicate the cell's leading pole, green arrows indicate how the cell moves during the tumble, pale grey cells show movement before the tumble, dark grey show the resulting motion.

- For a given rotation period, the mean rotation rate is less than 0.3 radians/min

More specifically, cells are attached to the surface at both poles, but these points of attachment move as the cell twiddles. Average cell speed in this context is linear speed as opposed to angular speed.

Automated and manual tracking data was collected from several experiments observing *P. aeruginosa* moving on a 2-dimensional surface. Automated data was tracked automatically using Fiji (ImageJ) and MATLAB and manual data was tracked by eye, both using bright-field microscopy videos of cells undergoing chemotaxis.

Four microfluidic channels were used to create concentration gradients of known chemoattractant DMSO, there being three different setups studied; control (no DMSO), DMSO gradient and an all DMSO control where the concentration of DMSO was uniform and non-zero. DMSO was added at 350 mM in all cases into relevant channels to create the appropriate gradients.

The experiments were conducted with wild type (WT) strains of *P. aeruginosa* as well as a mutant knockout strain, known here as ∇pilG, cells of which lack the protein pilG that allows bacteria to carry out the twitching motion we see in a twiddle (Buensuceso *et al.*, 2017). The ∇pilG strain thus carries out chemotaxis much less than the wild type.

The following experimental setups were conducted, all using different wild type strains, though considered here as experimental replicates for analyses, and those marked * were analysed:

- Experiment 1: allDMSOWT - DMSO gradient*, control*, DMSO control
- Experiment 2: ATCC vs Kolter - DMSO gradient* and control*
- Experiment 3: ∇pilG vs Wild Type (WT) - DMSO gradient*, control*, DMSO control

Names of individual strains are unimportant here and are only provided for consistency with the work of Wheeler (2020). Thus these replicates will be referred to as 'Experiment 1', 'Experiment 2' and 'Experiment 3' henceforth.

Both automated and manual data reported twiddle and reversal events for each cell track, with the direction of entry and exit being given in relation to the concentration gradient. A 'correct' entry or exit was considered to be in the direction of the highest concentration of DMSO, in the interval $(0, \pi)$, and a 'not correct' entry or exit was in the interval $[\pi, 2\pi]$. Though it may seem nonsensical to consider a correct entry in the absence of a gradient, we simply keep the terminology for consistency, in this scenario meaning cells were entering or exiting in the upwards direction. A visual representation of the movement of cells in a twiddle is shown in figure 4.2.

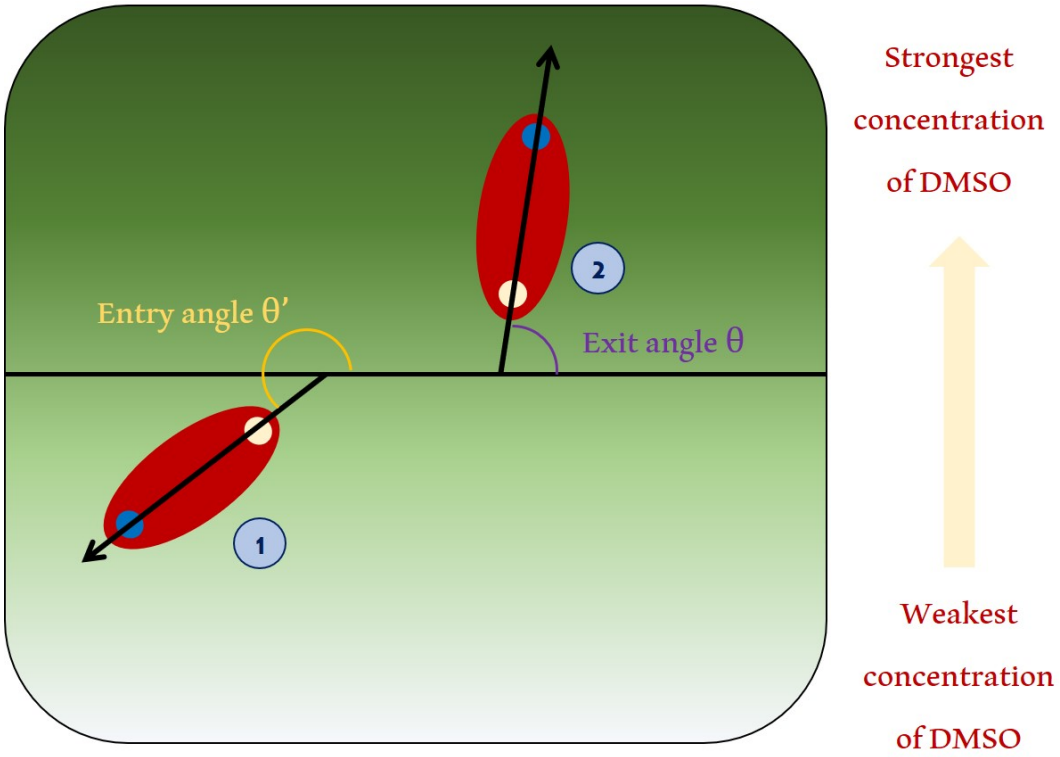

Figure 4.2: **Visual representation of entry into and exit from twiddles.** 1. The cell enters a twiddle at angle θ'. 2. The cell exits the twiddle at angle θ. Both θ' and θ are measured anticlockwise from the positive x-axis, and linear velocities are indicated by the arrows. The cell's poles are indicated by the circles at each end of the body, with the leading pole in blue. Correct entry or exit is an angle in the interval $(0, \pi)$, i.e. towards the strongest concentration of DMSO.

Reversals and twiddles could happen simultaneously: a cell can twiddle and

then reverse out of this twiddle, evidenced by the cell exiting the twiddle with the opposite leading pole to which it went in with. This is an important means of exit from a twiddle for a cell and such information was also recorded in the data as a binary variable with '1' meaning that a reversal occurred and '0' meaning it didn't. We can pool the data on WT bacteria across the three experiments when a gradient was used and in the case of the control.

Twiddles are often so long that the entries and exits cannot be paired reliably when tracking is carried out. Since the automated tracking algorithm used is more reliable at picking up entries than exits, a reverse time scale was used in the analysis so that twiddle entries correspond to twiddle exits and vice versa. There is however no pairing of entries into and exits from twiddles.

As explained above, entries and exits could be correct or not correct, though some of the data was recorded as 'na' meaning no data was recorded, or 'p' meaning the entry or exit was regarded as perpendicular to the direction of the gradient. Data was formatted so that p entries were regarded as not correct, na entries were not correct, na exits were treated as no data and the track was removed, and na reversals were treated as no reversal. This was decided upon so that correct exits and entries were definitely seen to be correct rather than misread or unknown. Observations recorded as p were chosen to be not correct as excluding them would potentially give results with false significance when testing bias hypotheses, and regarding them as correct was decided against in favour of needing correct entries or exits to be 'obviously' correct.

4.2.2 Data Analysis

It was decided that there were 3 key hypotheses to investigate before building the IBM.

1. Hypothesis 1: Are exits from twiddles equally likely to be correct or not correct?

2. Hypothesis 2: Are exits from twiddles independent of entries into them?

3. Hypothesis 3: Is exit from a twiddle independent of reversal within a twiddle, regardless of entry into the twiddle?

In order to be able to model chemotaxis with a Poisson process there would need to be independence of reorientation events, including independence of entry into and exit from the same twiddle since they will be considered separate events in the model (to allow for the cell to twiddle slowly over several time steps). Therefore these hypotheses should allow us to determine from the data whether or not the proposed IBM should be based around a Poisson process and give an idea of what other assumptions we can make, as well as giving an insight into the bias that twiddles cause.

Hypothesis 3, in particular, was chosen based on previous analysis of a similar kind on reversals alongside twiddles, and was intended to rigorously test and confirm previous conclusions. These were that reversals seemed to occur more often alongside twiddles after correct entry into a twiddle.

In all analyses in this section, hypotheses have been tested on the three experimental setups separately and then on the pooled data. Pooling was done by combining all data on WT strains from the three experiments; the data from the pilG mutant was not used. Both manual and automated data analyses are presented for comparison.

For ease of understanding, some notation is introduced for the following analyses.

- p_{C1} = proportion of correct exits in the absence of a gradient
- p_{G1} = proportion of correct exits in the presence of a gradient
- p_{C0} = proportion of not correct exits in the absence of a gradient
- p_{G0} = proportion of not correct exits in the presence of a gradient

Hypothesis 1

With this hypothesis we are looking at whether there is bias towards correct exits from twiddles. We first do this by comparing correct exits to not correct exits

in a gradient and then correct exits to not correct exits in the control. Then correct exits in a gradient are compared to correct exits in a control and as such hypothesis 1 is split into hypotheses 1a and 1b.

The statements of the hypotheses are

Hypothesis 1a

Null hypothesis: Correct and not correct exits from twiddles are equally likely in the absence (presence) of a gradient. $H_0 : p_{C1} = p_{C0}$ ($p_{G1} = p_{G0}$)

Alternative Hypothesis: Correct and not correct exits from twiddles are not equally likely in the absence (presence) of a gradient. $H_0 : p_{C1} \neq p_{C0}$ ($p_{G1} \neq p_{G0}$)

To be tested with a 2-sample chi-square proportion test using R prop.test.

Hypothesis 1b

Null hypothesis: Correct exits from twiddles are equally likely in the presence and absence of a gradient. $H_0 : p_{C1} = p_{G1}$

Alternative Hypothesis: Correct exits from twiddles are not equally likely in the presence and absence of a gradient. $H_0 : p_{C1} \neq p_{G1}$

To be tested with a 2-sample chi-square proportion test using R prop.test.

Beginning with hypothesis 1a, a 2-sample chi-square proportion test was carried out in R (`prop.test`, Statistics Package, (R Core Team, 2020)) to compare the correct and not correct exits in both gradient and control. This is done with all three possible alternative hypotheses: two-tailed, lower-tailed and upper-tailed, to determine how the proportions differ. When testing hypothesis 1b, again the 2-sample chi-square proportion test was again carried out in R. This will test whether the proportion of correct exits in the gradient is significantly different from the proportion of correct exits in the control, and again we test all three possible alternatives. The results obtained from conducting these hypotheses tests are shown in tables 4.1 (hypothesis 1a, manual), 4.2 (hypothesis 1a,

automated), 4.3 (hypothesis 1b, manual), and 4.4 (hypothesis 1b, automated), in the form p-value (conclusion). In all cases the significance level of 0.05 has been used when deciding whether to reject the null hypothesis.

Hypothesis 1a		
Manual Data	**Experiment 1**	**Experiment 2**
Control	23/67 = 0.3433 compared to 44/67 = 0.6567	35/76 = 0.4605 compared to 41/76 = 0.5395
Two-tailed	5.493e-04 ($p_{C1} \neq p_{C0}$)	0.4173 ($p_{C1} = p_{C0}$)
Upper-tailed	0.9927 ($p_{C1} = p_{C0}$)	0.7913 ($p_{C1} = p_{C0}$)
Lower-tailed	0.0002747 ($p_{C1} < p_{C0}$)	0.2087 ($p_{C1} = p_{C0}$)
Gradient	31/65 = 0.4769 compared to 34/65 = 0.5231	42/73 = 0.5753 compared to 31/73 = 0.4247
Two-tailed	0.7257 ($p_{G1} = p_{G0}$)	0.0979 ($p_{G1} = p_{G0}$)
Upper-tailed	0.6371 ($p_{G1} = p_{G0}$)	0.0489 ($p_{G1} > p_{G0}$)
Lower-tailed	0.3629 ($p_{G1} = p_{G0}$)	0.9511 ($p_{G1} = p_{G0}$)
	Experiment 3	**Pooled**
Control	37/81 = 0.4568 compared to 44/81 = 0.5432	95/224 = 0.4241 compared to 129/224 = 0.5759
Two-tailed	0.3458 ($p_{C1} = p_{C0}$)	0.0018 ($p_{C1} \neq p_{C0}$)
Upper-tailed	0.8271 ($p_{C1} = p_{C0}$)	0.9991 ($p_{C1} = p_{C0}$)
Lower-tailed	0.1729 ($p_{C1} = p_{C0}$)	9.1e-04 ($p_{C1} < p_{C0}$)
Gradient	42/81 = 0.5185 compared to 39/81 = 0.4815	115/219 = 0.5251 compared to 104/219 = 0.4749
Two-tailed	0.7533 ($p_{G1} = p_{G0}$)	0.3393 ($p_{G1} = p_{G0}$)
Upper-tailed	0.3767 ($p_{G1} = p_{G0}$)	0.1696 ($p_{G1} = p_{G0}$)
Lower-tailed	0.6233 ($p_{G1} = p_{G0}$)	0.8304 ($p_{G1} = p_{G0}$)

Table 4.1: Results for hypothesis 1a when tested on the manual data, formatted as p-value (conclusion). Proportions shown are the proportion of correct exits in each case. In all cases the significance level of 0.05 has been used when deciding whether to reject the null hypothesis.

Hypothesis 1a		
Automated Data	**Experiment 1**	**Experiment 2**
Control	74/142 = 0.5211 compared	51/90 = 0.5667 compared
	to 68/142 = 0.4789	to 39/90 = 0.4333
Two-tailed	0.6748 ($p_{C1} = p_{C0}$)	0.2463 ($p_{C1} = p_{C0}$)
Upper-tailed	0.3374 ($p_{C1} = p_{C0}$)	0.1231 ($p_{C1} = p_{C0}$)
Lower-tailed	0.6626 ($p_{C1} = p_{C0}$)	0.8769 ($p_{C1} = p_{C0}$)
Gradient	123/195 = 0.6308 compared	63/82 = 0.7683 compared
	to 72/195 = 0.3692	to 19/82 = 0.2317
Two-tailed	3.428e-04 ($p_{G1} \neq p_{G0}$)	2.049e-06 ($p_{G1} \neq p_{G0}$)
Upper-tailed	1.714e-04 ($p_{G1} > p_{G0}$)	1.024e-06 ($p_{G1} > p_{G0}$)
Lower-tailed	0.9998 ($p_{G1} = p_{G0}$)	1 ($p_{G1} = p_{G0}$)
	Experiment 3	**Pooled**
Control	49/113 = 0.4336 compared	174/345 = 0.5043 compared
	to 64/113 = 0.5663	to 171/345 = 0.4957
Two-tailed	0.1878 ($p_{C1} = p_{C0}$)	0.9143 ($p_{C1} = p_{C0}$)
Upper-tailed	0.9061 ($p_{C1} = p_{C0}$)	0.4571 ($p_{C1} = p_{C0}$)
Lower-tailed	0.0939 ($p_{C1} = p_{C0}$)	0.5429 ($p_{C1} = p_{C0}$)
Gradient	79/127 = 0.6220 compared	265/404 = 0.6559 compared
	to 48/127 = 0.3780	to 139/404 = 0.3441
Two-tailed	0.0078 ($p_{G1} \neq p_{G0}$)	5.004e-10 ($p_{G1} \neq p_{G0}$)
Upper-tailed	0.0039 ($p_{G1} > p_{G0}$)	2.502e-10 ($p_{G1} > p_{G0}$)
Lower-tailed	0.9961 ($p_{G1} = p_{G0}$)	1 ($p_{G1} = p_{G0}$)

Table 4.2: Results for hypothesis 1a when tested on the automated data, formatted as p-value (conclusion). Proportions shown are the proportion of correct exits in each case. In all cases the significance level of 0.05 has been used when deciding whether to reject the null hypothesis.

In the manual data sets, when looking at the proportion of correct exits compared to the proportion of not correct exits in the absence of a gradient, i.e.

Hypothesis 1b		
Manual Data	**Experiment 1**	**Experiment 2**
Control vs Gradient	$23/67 = 0.3433$	$35/76 = 0.4605$
	compared to	compared to
	$31/65 = 0.4769$	$42/73 = 0.5753$
Two-tailed	0.1663 ($p_{C1} = p_{G1}$)	0.2157 ($p_{C1} = p_{G1}$)
Upper-tailed	0.9169 ($p_{C1} = p_{G1}$)	0.8921 ($p_{C1} = p_{G1}$)
Lower-tailed	0.0832 ($p_{C1} = p_{G1}$)	0.1079 ($p_{C1} = p_{G1}$)
	Experiment 3	**Pooled**
Control vs Gradient	$37/81 = 0.4568$	$95/224 = 0.4241$
	compared to	compared to
	$42/81 = 0.5185$	$115/219 = 0.5251$
Two-tailed	0.5295 ($p_{C1} = p_{G1}$)	0.042 ($p_{C1} \neq p_{G1}$)
Upper-tailed	0.7352 ($p_{C1} = p_{G1}$)	0.979 ($p_{C1} = p_{G1}$)
Lower-tailed	0.2648 ($p_{C1} = p_{G1}$)	0.021 ($p_{C1} < p_{G1}$)

Table 4.3: Results for hypothesis 1b when tested on the manual data, formatted as p-value (conclusion). Proportions shown are the proportion of correct exits in each case. In all cases the significance level of 0.05 has been used when deciding whether to reject the null hypothesis.

hypothesis 1a, table 4.1 shows that for the Experiment 1 and Pooled cases we see evidence that these proportions are not equal, as we would expect. This is due to it being more likely that exits in the absence of a gradient are not correct given that cells recorded as having an exit perpendicular to the gradient are defined as not correct.

When testing the same proportions in a gradient, we see a significant result only in Experiment 2, where in the upper-tailed test we discover that the proportion of correct exits in a gradient is significantly greater than the proportion of not correct exits. These results are somewhat surprising given that we suspect there to be bias in exits in a gradient.

Looking at the results from testing the automated data in table 4.2, we see in every case that there is no bias for correct exits in the absence of a gradient

Hypothesis 1b		
Automated Data	**Experiment 1**	**Experiment 2**
Control vs Gradient	74/142 = 0.5211	51/90 = 0.5667
	compared to	compared to
	123/195 = 0.6308	63/82 = 0.7683
Two-tailed	0.0569 ($p_{C1} = p_{G1}$)	0.0085 ($p_{C1} \neq p_{G1}$)
Upper-tailed	0.9716 ($p_{C1} = p_{G1}$)	0.9958 ($p_{C1} = p_{G1}$)
Lower-tailed	0.0284 ($p_{C1} < p_{G1}$)	0.0042 ($p_{C1} < p_{G1}$)
	Experiment 3	**Pooled**
Control vs Gradient	49/113 = 0.4336	174/345 = 0.5043
	compared to	compared to
	79/127 = 0.6220	265/404 = 0.6559
Two-tailed	0.0053 ($p_{C1} \neq p_{G1}$)	3.72e-05 ($p_{C1} \neq p_{G1}$)
Upper-tailed	0.9974 ($p_{C1} = p_{G1}$)	1 ($p_{C1} = p_{G1}$)
Lower-tailed	0.0026 ($p_{C1} < p_{G1}$)	1.86e-05 ($p_{C1} < p_{G1}$)

Table 4.4: Results for hypothesis 1b when tested on the automated data, formatted as p-value (conclusion). Proportions shown are the proportion of correct exits in each case. In all cases the significance level of 0.05 has been used when deciding whether to reject the null hypothesis.

but there is a significant bias for correct exits compared to not correct exits in a gradient.

When testing hypothesis 1b, we see from table 4.3 for the manual data that the proportion of correct exits increases significantly in the presence of a gradient when observations are pooled, seemingly since we have close to significant results in the lower-tailed individual tests of hypothesis 1b for Experiment 1 and Experiment 2. This may be due to small sample sizes in the individual experiments meaning there is not enough power in the tests to reject the null hypothesis, though there is after pooling. It thus appears that there are significantly less correct exits in the absence of a gradient compared to in the presence of a gradient.

Testing of hypothesis 1b with the automated data concludes again in all cases that there are significantly less correct exits in the absence of the gradient than in the presence of one, as seen in table 4.4. This is what we would expect if twiddles are of benefit to bacteria travelling up a gradient by biasing their exits. We note that there is a lot more data here compared to the manual data sets, likely increasing the power of each individual test as well as the pooled ones.

Hypothesis 2

This hypothesis looks to determine if entries and exits associated with the same twiddle are independent. We then use the fact that if there truly is independence, we would expect the probability of one manner of exit to be similar to the other, regardless of manner of entry. For clarity, a more detailed explanation of how the chi-square test is used here and what it means in this context is provided in Appendix D. Only manual data is tested here as there is no pairing in the automated data set, rendering the notion of dependence nonsensical.

We can then formally state the hypotheses as:

Hypothesis 2

Null hypothesis: Exit direction is independent of entry direction.

Alternative hypothesis: Exit direction is not independent of entry direction.

To be tested using a chi-square test of independence (association) with SPSS Crosstabs analysis

The testing of hypothesis 2 was carried out on all experimental cases using a chi-square test of association with Crosstabs analysis in SPSS (IBM Corp, 2017). This tests whether or not two variables are independent of one another. The test statistic is computed based on the expected frequencies for the data we have and then a significant result is obtained if there is more deviation from these expected values than can be reasonably explained.

Sample sizes are given in appendix E and the results obtained from conducting these hypotheses tests are shown in table 4.5, in the form p-value (conclusion). In all cases the significance level of 0.05 has been used when deciding whether to reject the null hypothesis.

Hypothesis 2		
Manual Data	**Experiment 1**	**Experiment 2**
Control	0.086 (I)	0.343 (I)
Gradient	0.429 (I)	0.234 (I)
	Experiment 3	**Pooled**
Control	0.445 (I)	0.912 (I)
Gradient	0.765 (I)	0.659 (I)

Table 4.5: Results for hypothesis 2 when tested on the manual data, formatted as p-value (conclusion), I = independence, NI = no independence. In all cases the significance level of 0.05 has been used when deciding whether to reject the null hypothesis.

The results from the testing of manual data in table 4.5 show that exit is independent of entry in all cases, with the p-value for the Pooled case in the absence of a gradient providing the strongest evidence of independence. Overall, these results provide very strong evidence that going forward we can assume that entry into, and exit from twiddles are independent in all experimental cases and in both the presence and absence of a gradient.

Hypothesis 3

The final hypothesis investigates whether exit direction is independent of reversal within a twiddle, regardless of entry direction. This was intended to break down whether reversals cause bias in exits from twiddles and whether this was impacted by entry into the twiddle, as previous analysis conducted by Wheeler (2020) had shown that twiddles with correct entry saw a bias towards correct exit from a twiddle when a reversal and a twiddle occurred simultaneously. This hypothesis is tested by considering twiddle exit and reversal data, after twiddles are split by entry. The hypotheses to be tested are:

Correct entry

Null hypothesis: H_0 : When entering a twiddle correctly, exit direction is independent of reversal within the twiddle in the absence (presence) of a gradient

Alternative hypothesis: H_1 : When entering a twiddle correctly, exit direction is not independent of reversal within the twiddle in the absence (presence) of a gradient

Not correct entry

Null hypothesis: H_0 : When entering a twiddle not correctly, exit direction is independent of reversal within the twiddle in the absence (presence) of a gradient

Alternative hypothesis: H_1 : When entering a twiddle not correctly, exit direction is not independent of reversal within the twiddle in the absence (presence) of a gradient

To be tested using a chi-square test of association using SPSS Crosstabs analysis. Some cases require Fisher's exact test due to insufficient sample sizes.

To test these hypotheses, chi-square tests of association were carried out on numbers of exits and reversals after the entry split when all categories contained

at least 5 observations. When categories had less than 5 observations the chi-square test was no longer appropriate (McDonald, 2014), and so Fisher's exact test was used. In all cases a significance level of 0.05 was used when deciding on the conclusion of the hypothesis test. Sample sizes are given in appendix E and the results of the hypothesis tests are shown in table 4.6, formatted as p-value (conclusion). Again, only manual data is tested here as there is no pairing in the automated data set.

Hypothesis 3				
Manual Data	**Experiment 1**	**Experiment 2**	**Experiment 3**	**Pooled**
Control				
Not correct entry				
Chi-square	0.811 (I)	0.132 (I)	0.051 (I)	0.045 (NI)
Fisher's	0.565 (I)	0.191 (I)	0.192 (I)	na
Correct entry				
Chi-square	0.885 (I)	0.457 (I)	0.047 (NI)	0.687 (I)
Fisher's	1 (I)	0.624 (I)	0.222 (I)	na
Gradient				
Not correct entry				
Chi-square	0.757 (I)	0.047 (NI)	0.757 (I)	0.207 (I)
Fisher's	1 (I)	0.084 (I)	1 (I)	na
Correct entry				
Chi-square	0.017 (NI)	0.025 (NI)	0.00 (NI)	0.00 (NI)
Fisher's	0.030 (NI)	0.041 (NI)	na	na

Table 4.6: Results for hypothesis 3 when tested on the manual data, formatted as p-value (conclusion), I = independence, NI = no independence. In all cases the significance level of 0.05 has been used when deciding whether to reject the null hypothesis.

Results from the manual analyses in table 4.6 show that in the absence of a gradient we largely see that exits from twiddles are independent of reversals, except for when we have not correct entry in the Pooled case and correct entry in the Experiment 3 case. In a gradient, we see that all tests report non-independence,

meaning that there is strong evidence to show that exits from twiddles are biased when they happen with a reversal if the entry to the twiddle is correct. These findings confirm the suspicions that arose from preliminary analysis, that we see some bias for correct twiddle exit when there is correct entry and simultaneous reversal.

This could suggest that if bacteria enter a twiddle correctly and reorient by $\pi(+2\pi n)$, they are biasing their exit by reversing after twiddling so that they end up once again in the correct direction. No data on the number of revolutions cells undergo during a twiddle was available, but data of this kind would allow a clearer picture of turn angle distributions and would allow further investigation of this hypothesis. Correlation between entry to and exit from a twiddle could also be further studied with this kind of data.

Bar charts for hypothesis 3 For a visual reference, bar charts of the proportions considered in hypothesis 3 are given for the manual data sets. These charts compare the proportion of correct exits for all cells with the proportion of correct exits when the twiddle was not accompanied by a reversal, and this is done for control data and gradient data separately. Error bars representing the 95% confidence interval for the estimate of the proportion are given, calculated using the 1-sample proportion test in R. We are looking to see if there is a significant difference in these proportions, i.e. that the confidence intervals do not overlap.

Looking first at the bar charts for correct entry, we see that in figure 4.3 there are no significant differences in the proportion of correct exits between all twiddles and when simultaneous reversals are excluded. The closest to a significant difference is seen in the Pooled case, in figure 4.3d), when comparing the proportions of correct exits in a gradient. It looks as if there are almost significantly more correct exits when reversals occur simultaneously than when these twiddles are excluded, and the confidence intervals themselves show that there is only an overlap of 0.02.

The plots for not correct entry in figure 4.4 also show no significant differences in proportions of correct exits when reversals are excluded compared to when

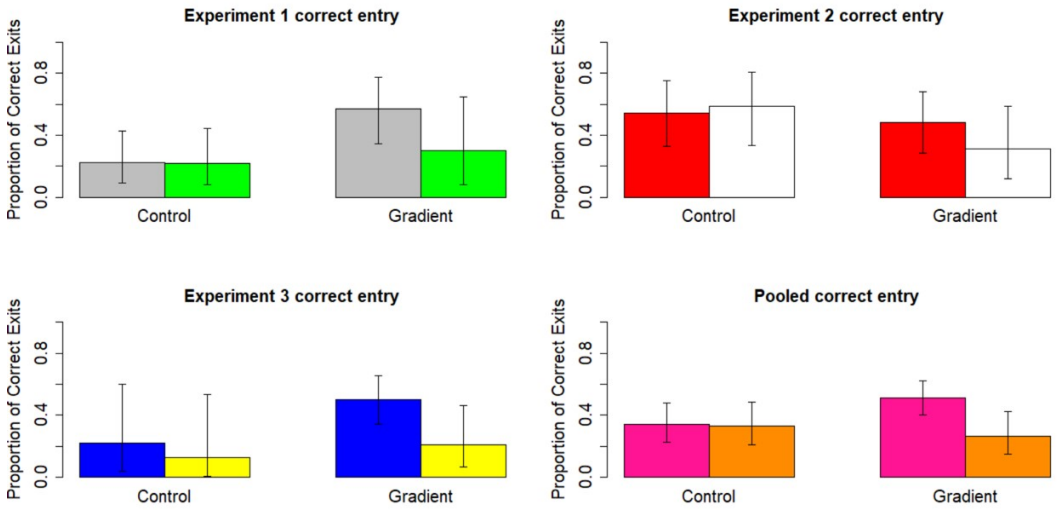

Figure 4.3: **Bar plots showing the proportions of correct exits after correct entry into twiddles for all experimental cases.** Plots show the proportions of correct exits when all twiddles are considered compared to when twiddles with simultaneous reversal are excluded, after correct entry into a twiddle. Comparison is made between twiddles in the presence of gradient and the absence of a gradient (control). **a)** Plots for Experiment 1 with all twiddles in grey, reversals excluded in green. **b)** Plots for Experiment 2 with all twiddles in red, reversals excluded in white. **c)** Plots for Experiment 3 with all twiddles in blue, reversals excluded in yellow. **d)** Plots for the Pooled case with all twiddles in pink, reversals excluded in orange.

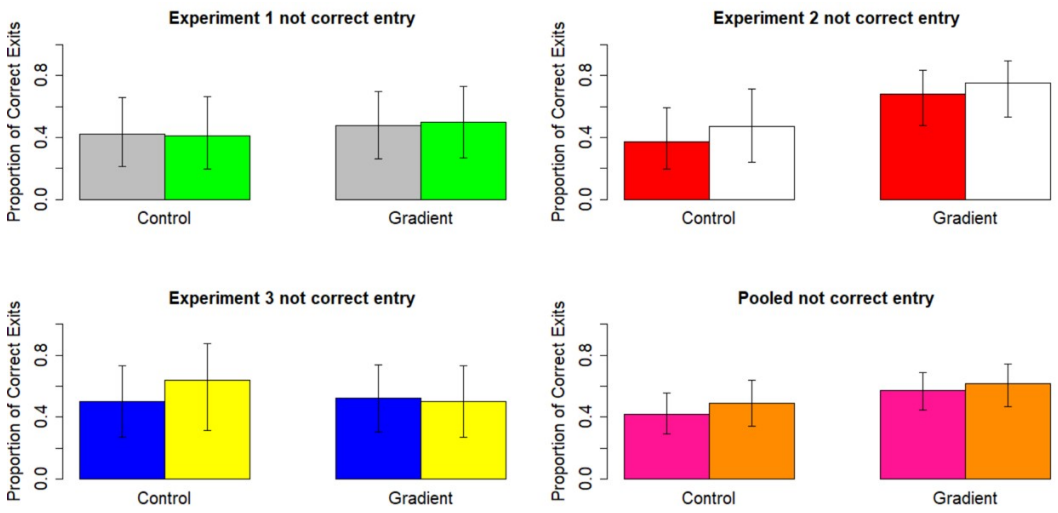

Figure 4.4: **Bar plots showing the proportions of correct exits after not correct entry into twiddles for all experimental cases.** Plots show the proportions of correct exits when all twiddles are considered compared to when twiddles with simultaneous reversal are excluded, after not correct entry into a twiddle. Comparison is made between twiddles in the presence of gradient and the absence of a gradient (control). **a)** Plots for Experiment 1 with all twiddles in grey, reversals excluded in green. **b)** Plots for Experiment 2 with all twiddles in red, reversals excluded in white. **c)** Plots for Experiment 3 with all twiddles in blue, reversals excluded in yellow. **d)** Plots for the Pooled case with all twiddles in pink, reversals excluded in orange.

they are not. Interestingly though, in almost every case the proportion of correct exits is higher when there are no reversals in both the control and the gradient, suggesting that after a not correct entry, reversals may reduce the number of correct exits from twiddles. As this bias seems to be present in both the control and gradient though, it may be that this effect of reversals is always present and isn't affecting the twiddle exit.

In all cases where there has been correct entry (figure 4.3), we see a much lower proportion of correct exits when we exclude reversals in a gradient. The small sample sizes in this data set are likely giving wide confidence intervals and affecting the power of the hypothesis test, as mentioned above, thus not allowing us to definitively say that simultaneous reversals increase the proportion of correct exits. With more data though, it looks likely that we would conclude that reversals alongside twiddles are beneficial after a correct entry into the twiddle. This is supported by the fact that the Pooled case is almost significant. This conclusion is in agreement with the testing of hypothesis 3 and could be helpful to get the cell back onto the correct track.

On the whole the results are varied and so make it hard to come to any solid conclusions surrounding hypothesis 3 from the visualisation alone. The results for individual experiments are also somewhat varied and it can be difficult to tell if this is down to experimental error or differences in twiddle behaviour between strains. This all means that these results should be taken with caution in cases where we see this variation between experiments.

4.2.3 Individual-Based Model

We want to create an IBM to simulate cell tracks of surface-attached bacteria which can ultimately investigate the impact of reversals, twiddles and both together on mean chemotactic drift. To achieve this we will use the results of the hypothesis tests above, namely hypothesis 1b which confirms that there is bias for correct exits in a gradient, and hypothesis 2 confirming that entry and exit angle are independent. We will also assume that reorientations occur instantaneously, to simplify the initial model. We will use Poisson processes to model

reversals and the entry into and exit from twiddles, creating separate IBMs for reversals and twiddles, and assuming the turn rate depends on the angle of orientation. In order to explain the IBM and what it does we must first look to the mathematical model on which it will be based, taken from Bearon & Durham (2019).

Velocity Jump Model

The model introduced by Bearon & Durham (2019), is a velocity jump process for both weak and strong chemotaxis. It looks at how strategies of motion combined with either type of chemotaxis are beneficial for bacteria in different environments.

The conservation equation for the probability distribution function $\psi(\boldsymbol{x}^*, \boldsymbol{p}, t^*)$ which represents the distribution of cells with position $\boldsymbol{x}^*$ and direction of movement $\boldsymbol{p}$ at time t^* is given by

$$\frac{\partial \psi}{\partial t^*} + \boldsymbol{\nabla}_{\boldsymbol{x}^*} \cdot (V_s \boldsymbol{p} \psi) + \lambda^* \psi - \int_\Omega \lambda^*(\boldsymbol{p'}) K(\boldsymbol{p}, \boldsymbol{p'}) \psi(\boldsymbol{p'}) d\boldsymbol{p'} = 0, \tag{4.1}$$

where $*$ represents a dimensional quantity and functions are assumed to be evaluated at $(\boldsymbol{x}^*, \boldsymbol{p}, t^*)$ unless otherwise stated. In this equation $\boldsymbol{p'}$ and $\boldsymbol{p}$ are the directions of movement before and after reorientation events, and V_s is the speed of cells, meaning that velocity is given by $V_s \boldsymbol{p'}$. Cells turn away from direction $\boldsymbol{p'}$ with frequency $\lambda^*(\boldsymbol{p'})$, choosing a new direction $\boldsymbol{p}$ with probability $K(\boldsymbol{p}, \boldsymbol{p'})$. Turn kernel K captures reorientations, giving the conditional probability of an exit direction $\boldsymbol{p}$ given an entry direction of $\boldsymbol{p'}$. A Poisson process is assumed here, so that the turn rate is independent of the run time.

The turn rate, λ can be written as

$$\lambda^* = \lambda_0 \exp(-\zeta V_s \boldsymbol{p} \cdot \boldsymbol{\nabla} \boldsymbol{s}), \tag{4.2}$$

for basal turn rate λ_0 in the absence of a gradient, chemoattractant concentration s and ζ being proportional to $K_D/(K_D+s^2)$, for dissociation constant K_D, related to the amount of chemoattractant bound to the cell (Brown & Berg, 1974).

This means that cells moving up a chemical gradient reduce their turn rate as more chemoattractant binds to their receptors, and cells moving down the

gradient will increase their turn rate. This strategy is known as bi-bias (or bi-directional bias).

We will be considering chemotaxis on a surface, thus must constrain cells to a 2-dimensional plane with a gradient increasing in the positive y^*-direction. Therefore we introduce direction vector $\boldsymbol{p}$ in terms of the angle θ as

$$\boldsymbol{p} = \cos\theta \boldsymbol{i} + \sin\theta \boldsymbol{j}.$$

The governing equation then becomes an equation for $\psi(y^*, \theta, t^*)$ and similarly to 4.1, is given by

$$\frac{\partial \psi}{\partial t^*} + \frac{\partial}{\partial y^*}(V_s \sin\theta \psi) + \lambda^* \psi - \int_0^{2\pi} \lambda^*(\theta') K(\theta, \theta') \psi(\theta') d\theta' = 0, \tag{4.3}$$

where turn kernel $K(\theta, \theta')$ is the conditional probability of having exit angle θ given the entry angle was θ'.

We then rewrite the turn rate 4.2 as

$$\lambda^* = \lambda_0 \exp\left(-\zeta V_s \sin\theta \frac{ds}{dy^*}\right).$$

If we let $\chi = -\zeta \frac{ds}{dy^*}$, then the turn rates can be finally written as

$$\lambda^* = \lambda_0 \exp\left(-V_s \chi \sin\theta\right). \tag{4.4}$$

It will also be useful for us to consider the steady direction distribution that cells will reach at equilibrium i.e. the steady, spatially homogeneous solution to equation 4.3, where ψ doesn't vary with y^* or t^*. We define $f_E(\theta)$ as the steady direction distribution representing an equilibrium distribution where cells turn away from angle θ at the same rate as they turn towards it, which satisfies

$$\lambda(\theta) f_E(\theta) - \int_0^{2\pi} \lambda(\theta') K(\theta, \theta') f_E(\theta') d\theta' = 0. \tag{4.5}$$

Reversal only model

We first start by outlining the relevant expressions for modelling reversals under the framework above.

The turn kernel for reversals can be written as

$$K(\theta, \theta') = \delta(|\theta - \theta'| - \pi) = \begin{cases} 1, & |\theta - \theta'| = \pi, \\ 0, & \text{otherwise}, \end{cases} \tag{4.6}$$

for a cell turning from its entry angle θ' to its exit angle θ, as a cell can either reverse and change its direction by π, or only moves a small amount due to rotational diffusion. This ensures that the cell will choose to reorient by π if it reverses, due to K having value 1 when the magnitude of the turn is equal to π. We also note that the turn kernel here depends on $|\theta - \theta'|$, as in Bearon & Durham (2019).

We can also look at the equilibrium direction distribution for reversals, $f_E^R(\theta)$. Recall that this must solve equation 4.5, and so we obtain

$$f_E^R(\theta) = \frac{1}{\int_0^{2\pi} \frac{1}{\lambda(\theta)} d\theta} \frac{1}{\lambda(\theta)}, \tag{4.7}$$

where $\int_0^{2\pi} f_E^R(\theta)\, d\theta = 1$. It is important to note that this equilibrium solution is only reached for a smooth turn kernel, that could be the delta function with some rotational noise added. Equation 4.7 is the unique equilibrium orientation distribution for reversals which is reached over time, providing that either initial conditions are defined and the delta function is used as the turn kernel, or that rotational diffusion is added to smooth out the delta function acting as the turn kernel. In this work we add rotational diffusion as a parameter to simulations, thus choosing the latter. More details on how equation 4.7 forms the solution to the system created by equations 4.5 and 4.6 are given in appendix F.

We are interested in what happens to the number of turns at steady state. The number of turns from some interval $[\theta, \theta + d\theta]$ in time interval $[t, t + dt]$ is given by $dt \times d\theta \times$ turn rate $\times$ number of cells swimming in direction θ, i.e. $dt \times d\theta \times \lambda(\theta) \times f(\theta) \times$ Ncells, for total number of cells, Ncells, and the direction distribution $f(\theta)$ at time t. We here neglect spatial variation in cell density, so the total number of cells with orientation θ is just the total number of cells in the experiment multiplied by $f(\theta)$.

At steady state $f(\theta)$ becomes $f_E^R(\theta)$ and so we have just $dt \times d\theta \times$ Ncells as the number of turns both away from and toward some interval $[\theta, \theta + d\theta]$, but

this number is independent of θ, i.e. we should see a uniform number of turns in all directions at steady state.

For any particular interval $[\theta, \theta + d\theta]$, when we count the number of turns away from and toward it in time interval $[t, t + dt]$, we should get approximately the same number. Extrapolating this, at steady state we should see an equal number of cells turning towards the strongest concentration in the gradient as away from it.

In the context of the experiments analysed earlier in this chapter, we should see as many cells swimming up the gradient and reversing as swimming down the gradient and reversing. This is therefore an important observation about any experimental data sets we study and in determining if the model from Bearon & Durham (2019) can be used to model these cells. We expect that after period of time long enough for the steady direction distribution to have been reached, we would see uniform numbers of reversals both towards and away from the direction of strongest concentration.

Twiddle only model

Doing the same again, we can define the turn kernel for twiddles, this time introducing an exit bias such that exits are biased towards the direction of strongest concentration of chemoattractant, $\pi/2$. We here use the information from hypothesis 1 that there is a bias for correct exits and information from hypothesis 2 that exit and entry are independent. Entry angles are uniform over the interval $[0, 2\pi]$ and exit angles are defined by the von Mises distribution centred on $\pi/2$. Since entry and exit are independent, the probability of a certain pair of exit and entry angles is found by multiplying the respective density functions, so the conditional probability given by the turn kernel can be written as

$$K(\theta, \theta'; \kappa) = \frac{1}{2\pi I_0(\kappa)} e^{\kappa \cos(\theta - \pi/2)}, \tag{4.8}$$

where I_0 is the modified Bessel function of order 1. κ is the concentration parameter of the von Mises distribution. In this context, when $\kappa = 0$ we have isotropic turns i.e. no bias in exit angle. As κ increases the bias for exit from twiddles

towards $\pi/2$ also increases. We note that this turn kernel only depends on exit angle θ.

We also look again at the equilibrium direction distribution, this time for twiddles, denoted $f_E^T(\theta)$. Again, this distribution must satisfy equation 4.5, though this time we note that there is no dependence on θ' in the turn kernel K, and so we can obtain $f_E^T(\theta)$ as follows

$$\begin{aligned}
&\lambda(\theta) f_E^T(\theta) = \int_0^{2\pi} \lambda(\theta') K(\theta, \theta') f_E^T(\theta') d\theta' = cK(\theta) \\
\implies &f_E^T(\theta) = \frac{cK(\theta)}{\lambda(\theta)} = \frac{c}{2\pi I_0(\kappa)\lambda(\theta)} e^{\kappa \cos(\theta - \pi/2)} \\
\implies &f_E^T(\theta) = A \frac{e^{\kappa \cos(\theta - \pi/2)}}{\lambda(\theta)} \propto \frac{e^{\kappa \cos(\theta - \pi/2)}}{\lambda(\theta)}, \qquad (4.9)
\end{aligned}$$

for $A = \frac{c}{2\pi I_0(\kappa)}$ which varies with κ, and $c = \int_0^{2\pi} \lambda(\theta') f_E^T(\theta')\, d\theta'$ which is a function only of θ' and not θ.

Following a similar logic to above, this version of f_E suggests that the number of turns away from θ, $f(\theta) \times \lambda(\theta)$ is not uniform, but depends on the turn kernel K. This means that variation in $\lambda(\theta)$ does not affect the number of turns away from θ, but having a bias in exits does. So when there is an exit bias we would expect to see a bias in the number of turns away from angle θ, in contrast to the uniform number we expect in the reversals only model. Translating this into the current context, we do not expect to see the same number of cells swimming up the gradient and twiddling as those swimming down and twiddling.

Simulation

With the mathematical framework in place, we can build individual-based simulations using this model and the turn rates and parameters given above.

Parameters We first introduce the parameters used in the simulations, with table 4.7 summarising the details each of them. In each case there is a basal turn rate λ_{R_b} or λ_{T_b}, along with a chemotactic strength or bias χ_R or χ_T which affect rate of entry into these turns.

The simulations are run with a number of cells, Ncells, between times t_0 and

t_N with time step dt. We also use cell speed V_s and rotational diffusion coefficient D_r.

Parameter	**Description**
λ_{R_b}	Basal reversal rate
χ_R	Chemotactic bias in reversal rate
λ_{T_b}	Basal twiddle rate
χ_T	Chemotactic bias in twiddle entry rate
κ	Bias in exit from a twiddle
D_r	Rotational diffusion coefficient
V_s	Cell speed
Ncells	Number of cells in simulation
dt	Length of time step
t_0	Start time of simulation
t_N	End time of simulation
T	Total simulation time

Table 4.7: Parameters involved in the individual-based simulations for reversals and twiddles.

The simulation takes the following form.

- Give a cell an orientation and position, initial positions being at the origin and initial orientation being rand $\times 2\pi$, where rand is a random number sampled from Unif$(0, 1)$. Starting cells at the same point will give a clearer picture of the bias seen in the tracks, but the choice of initial position is arbitrary

- In each time step, update the velocity and position according to the following:

  - Sample p from Unif$(0, 1)$

  - If $p < \lambda^*$, where λ^* is as defined in equation 4.4, then the cell turns. If this is the case then the current position is plotted as a star on the track to signify a turn has occurred, the orientation is updated and

the velocity and position are updated as

$$\begin{aligned} v_{xi} &= V_s \cos\theta_i \\ v_{yi} &= V_s \sin\theta_i \\ x_i &= x_{i-1} + v_{xi}\, dt \\ y_i &= y_{i-1} + v_{yi}\, dt. \end{aligned}$$

- Orientation θ is updated as

$$\theta_i = \theta_i^* + \text{randn}_i \sqrt{2D_r dt},$$

  which adds rotational noise to the updated angle θ_i^*, the angle modified by the reversal or twiddle, as explained below, and randn here is a random number sampled from the standard normal distribution.

Reversal only model For reversals alone, the turn rate is given by equation 4.4 with chemotactic bias given by χ_R and basal turn rate λ_0 equal to λ_{R_b}. Updating a cell's orientation after a reversal would look like

$$\theta_i^* = \theta_{i-1} + \pi,$$

which will reverse the direction of the cell.

Example visual output for the Reversal only model is shown in figure 4.5 for arbitrary parameter values to demonstrate the IBM. It is seen from figure 4.5b) that when the turn rate is as in equation 4.4 and includes the chemotactic bias parameter χ_R there is a definite drift in the tracks up the gradient, compared to when turn rate is basal in figure 4.5a). We see from figure 4.5c) that the number of reversals away from angle θ is roughly uniform as would be expected when steady state has been reached, and figure 4.5d) shows that the distribution of $f(\theta)$ at the end of the simulation follows the expected theoretical distribution as given by the continuum model. Figure 4.6 shows how the chemotactic drift velocity, v_D changes with chemotactic bias in the reversal rate, χ_R. We see that as χ_R increases, the drift velocity is also increased.

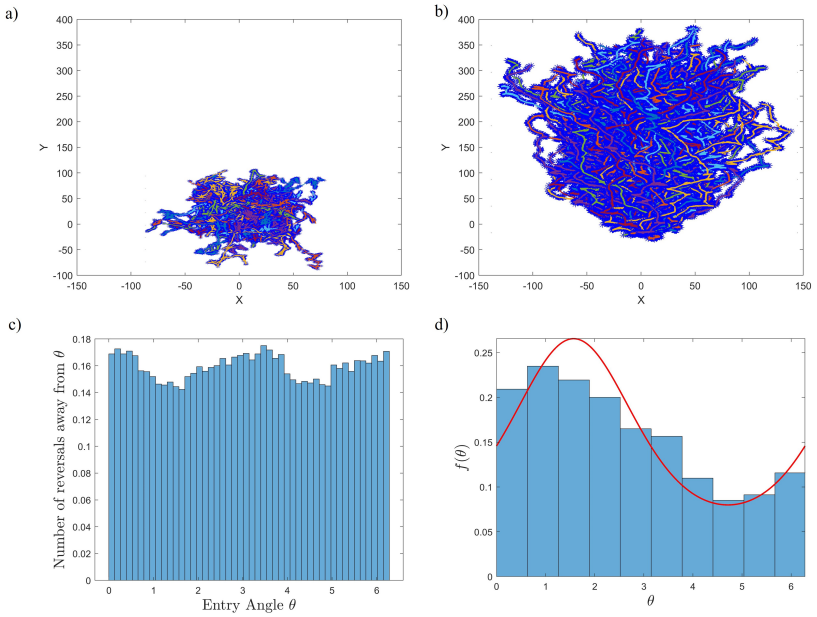

Figure 4.5: **Visual output from IBM simulations in the Reversals only model.** Plots are shown for simulations with parameters Ncells $= 200$, $T = 1000$, $dt = 0.1$, $D_r = 0.01$, $V_s = 1$, $\lambda_{R_b} = 0.6$. **a)** Plot of cell tracks with no chemotactic bias, i.e. $\chi_R = 0$. **b)** Plot of cell tracks with chemotactic bias $\chi_R = 0.6$. **c)** Histogram of number of reversals away from angle θ, for $\chi_R = 0.6$. **d)** Histogram of simulated values of $f(\theta)$ for $\chi_R = 0.6$ using data from the last 100 time steps, with the theoretical $f_E^R(\theta)$ as in equation 4.7, overlaid in red. All histograms have been normalized so that the area covered by the bars is less than or equal to 1 and thus the plot is an estimate of the probability density function.

Twiddle only model For twiddles alone, the turn rate is again given by equation 4.4 but with chemotactic bias given by χ_T and basal turn rate λ_0 equal to λ_{T_b}. Updating a cell's orientation after a twiddle would look like

$$\theta_i^* = \text{randvM}_i,$$

which will choose a randomly sampled exit angle from the unit circle weighted by the von Mises distribution centred at $\pi/2$, the direction of the strongest concentration of chemoattractant, as described in equation 4.8.

Example visual output for the Twiddle only model is shown in figure 4.7 for arbitrary parameter values to demonstrate the IBM. We again see bias up

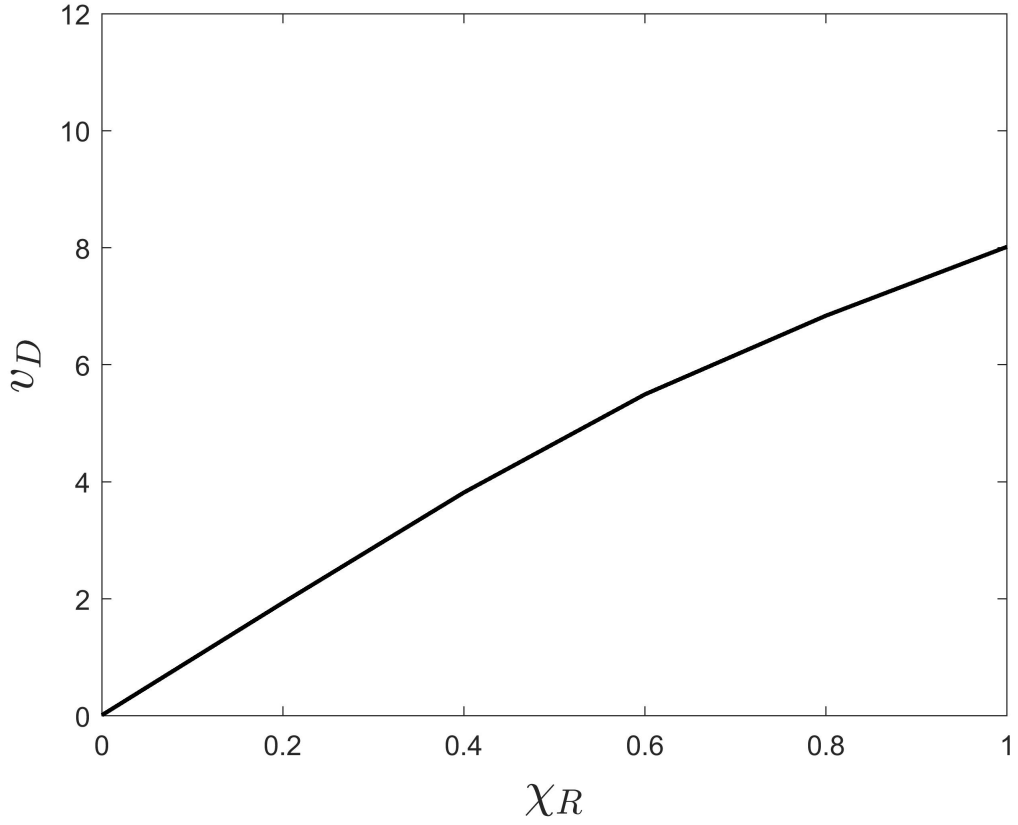

Figure 4.6: **Plot showing how chemotactic drift velocity v_D changes with chemotactic bias in reversal rate, χ_R, in the Reversal only model.** Plots are shown for the simulations with parameters Ncells = 200, $T = 1000$, $dt = 0.1$, $D_r = 0$, $V_s = 1$, $\chi_R = 0.6$, $\lambda_{R_b} = 0.6$.

the gradient in the tracks in figure 4.7b) where turn rate includes chemotactic bias through parameter χ_T, compared to the basal turn rate in figure 4.7a). Figure 4.7c) this time shows that the number of twiddles away from θ is not uniform as was the case for reversals, but follows a von Mises distribution as expected. Figure 4.5d) shows that the theoretical $f_E^T(\theta)$ describes the simulated values of $f(\theta)$ at the end of the simulation fairly well. All simulation code can be found at the link given in Appendix G.

Figure 4.8 shows how the chemotactic drift velocity, v_D changes with chemotactic bias in the twiddle rate, χ_T. We see again that as χ_T increases, the drift velocity increases, though the velocity is higher overall compared to that of reversals. This is what we would expect due to twiddles having a greater biasing

effect on the chemotactic motion and thus causing displacement to be increased over the simulation when cells are twiddling.

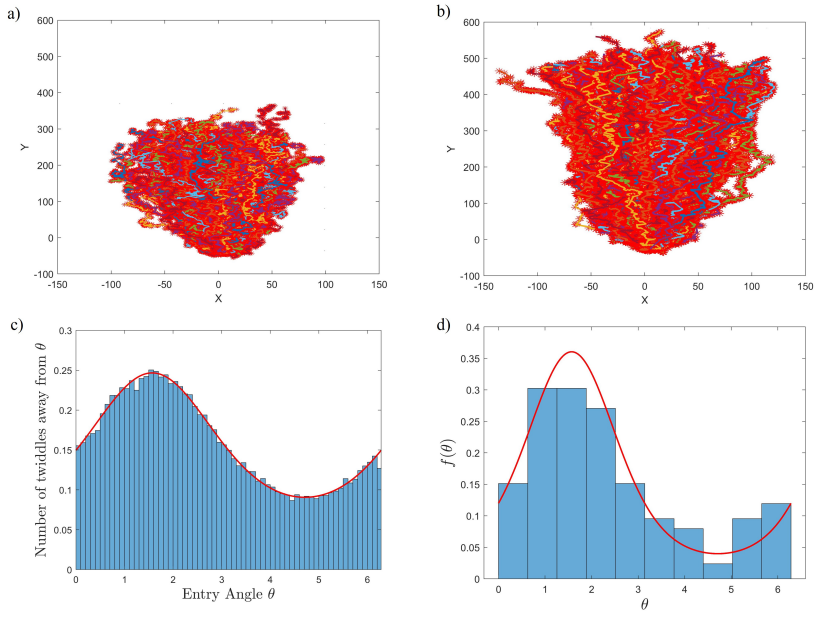

Figure 4.7: **Visual output from IBM simulations in the Twiddles only model.** Plots are shown for simulations with parameters Ncells $= 200$, $T = 1000$, $dt = 0.1$, $D_r = 0$, $V_s = 1$, $\chi_T = 0.6$, $\kappa = 0.5$, $\lambda_{T_b} = 0.6$. **a)** Plot of cell tracks with no chemotactic bias i.e. $\chi_T = 0$. **b)** Plot of cell tracks with chemotactic bias $\chi_T = 0.6$. **c)** Histogram of number of twiddles away from angle θ for $\chi_T = 0.6$, with the overlaid red line showing the predicted number of turns away from angle θ, proportional to the von Mises density function with parameter $\pi/2$. **d)** Histogram of simulated values of $f(\theta)$ for each value of theta at the end of the simulation for $\chi_T = 0.6$, with the theoretical $f_E^T(\theta)$ overlaid in red. All histograms have been normalized so that the area covered by the bars is less than or equal to 1 and thus the plot is an estimate of the probability density function.

Obtaining realistic parameter values from experimental data In order to estimate parameters from experimental data we need to consider what we can get from the data we have. In this case we are limited to knowledge of turns, both reversals and twiddles, that are correct and not correct, i.e. up or down. We will define 'up' in relation to a cell that is travelling up the gradient

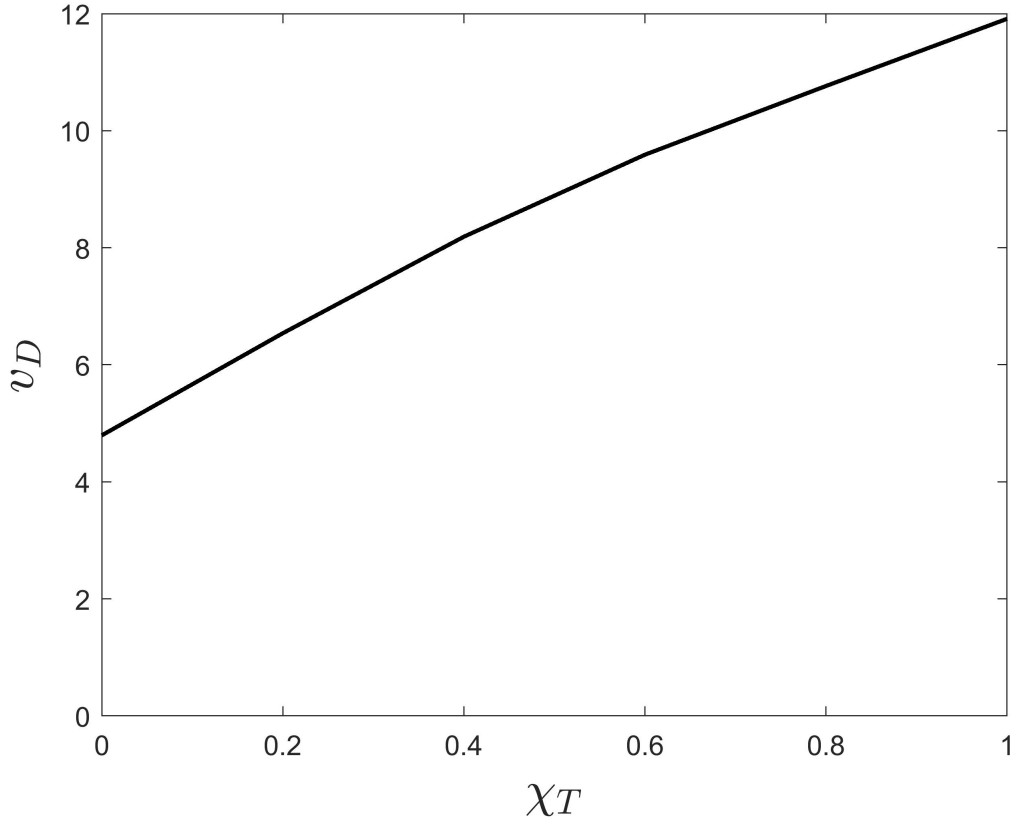

Figure 4.8: **Plot showing how chemotactic drift velocity v_D changes with chemotactic bias in twiddle rate, χ_T, in the Twiddle only model.** Plots are shown for the simulations with parameters Ncells = 200, $T = 1000$, $dt = 0.1$, $D_r = 0$, $V_s = 1$, $\chi_T = 0.6$, $\kappa = 0.5$, $\lambda_{T_b} = 0.6$.

with orientation in $(0,\ \pi)$ and turns, which is the same as correct entry, and 'down' in relation to a cell that is travelling down the gradient with orientation in $[\pi,\ 2\pi]$, (including parallel entries) and turns, so the same as not correct entry.

The extraction of realistic parameter values from the data using the parameters as explained in Bearon & Durham (2019) is now demonstrated. First, recall that turns occur at rate $\lambda^*(\theta) = \lambda_0 \exp(-V_s \chi \sin\theta)$ using the bi-bias strategy, where λ_0 is the basal turn rate, V_s is the cell speed, χ is a measure of chemotactic strength and θ is the angle with which a cell enters a turn. Stars will be omitted hence forth, though the meaning remains the same.

The average rate of turns occurring when cells are travelling up the gradient

can be written as

$$\lambda_{up} = \int_0^{\pi} \lambda(\theta) f_{E_{up}}(\theta) d\theta,$$

where $f_{E_{up}}(\theta)$ is the steady direction distribution for cells travelling up the gradient, i.e. $f_{E_{up}}(\theta) = \frac{1}{\int_0^{\pi} \frac{1}{\lambda(\theta)} d\theta} \frac{1}{\lambda(\theta)}$ for reversals and $f_{E_{up}}(\theta) = \frac{\int_0^{\pi} \lambda(\theta') f_E^T(\theta')\, d\theta'}{2\pi I_0(\kappa)\lambda(\theta)} e^{\kappa \cos(\theta - \pi/2)} = A \frac{e^{\kappa \cos(\theta-\pi/2)}}{\lambda(\theta)}$ for twiddles.

To estimate χ_R and λ_{R_b} for the reversal only model we can first write

$$\lambda_{up} = \int_0^{\pi} \frac{1}{\int_0^{\pi} \frac{1}{\lambda(\theta)} d\theta} d\theta = \frac{\pi}{\int_0^{\pi} \frac{1}{\lambda(\theta)} d\theta} = \frac{\pi \lambda_{R_b}}{\int_0^{\pi} e^{V_s \chi_R \sin\theta} d\theta}. \tag{4.10}$$

This also tells us that a similar expression for λ_{down} will be

$$\lambda_{down} = \frac{\pi \lambda_{R_b}}{\int_{\pi}^{2\pi} e^{V_s \chi_R \sin\theta} d\theta} \tag{4.11}$$

We can also get estimates of these quantities from the experimental data, let's call them λ_{up}^D and λ_{down}^D, so that

$$\lambda_{up}^D = \frac{\text{Number of turns occurring when cells travel up the gradient}}{\text{Number of cells travelling up the gradient} \times \text{time}}$$

and

$$\lambda_{down}^D = \frac{\text{Number of turns occurring when cells travel down the gradient}}{\text{Number of cells travelling down the gradient} \times \text{time}}.$$

The 'time' referred to in these formulae is some experimental time over which we can assume that only one reorientation event occurs in each cell. For twiddles, this is likely to be the whole experiment i.e. over 300 minutes, but for reversals this is likely to be a shorter time interval.

We can then look to the ratio of $\lambda_{up}/\lambda_{down}$, which analytically is given by

$$\frac{\lambda_{up}}{\lambda_{down}} = \frac{\frac{\pi \lambda_{R_b}}{\int_0^{\pi} e^{V_s \chi_R \sin\theta} d\theta}}{\frac{\pi \lambda_{R_b}}{\int_{\pi}^{2\pi} e^{V_s \chi_R \sin\theta} d\theta}} = \frac{\int_{\pi}^{2\pi} e^{V_s \chi_R \sin\theta} d\theta}{\int_0^{\pi} e^{V_s \chi_R \sin\theta} d\theta}.$$

Once we have this formula, we can then carry out numerical integration over a range of different χ_R values, calculate the ratio of $\lambda_{up}/\lambda_{down}$ and compare this to the ratio of $\lambda_{up}^D/\lambda_{down}^D$. The value of χ_R for which the margin of error between these two ratios is smallest will be chosen as the realistic parameter value for χ_R. 'Smallest' here is defined as $\min(|\lambda_{up}^D/\lambda_{down}^D - \lambda_{up}/\lambda_{down}|)$. The chosen value

of χ_R can then be substituted into either equation 4.10 or 4.11 to estimate the realistic value of λ_{R_b}.

Using data from the Reversal Only model simulations as an example, when $\lambda_{R_b} = 0.6, \chi_R = 1, dt = 0.1, \text{Ncells} = 200, t_N = 1000$, the simulations gave a value for $\lambda_{up}/\lambda_{down} = 0.3716/0.7688 = 0.4834$. Upon using these values in the analytic expressions, λ_{R_b} was estimated as 0.5433.

For the twiddle only model, expressions for λ_{up} and λ_{down} are given by

$$\lambda_{up} = A \int_0^{\pi} e^{\kappa \cos(\theta - \pi/2)}\, d\theta$$
$$\lambda_{down} = A \int_{\pi}^{2\pi} e^{\kappa \cos(\theta - \pi/2)}\, d\theta,$$

meaning that the ratio of interest becomes

$$\frac{\lambda_{up}}{\lambda_{down}} = \frac{\int_0^{\pi} e^{\kappa \cos(\theta - \pi/2)}\, d\theta}{\int_{\pi}^{2\pi} e^{\kappa \cos(\theta - \pi/2)}\, d\theta}.$$

Thus we cannot estimate corresponding parameters λ_{T_b} and χ_T using this ratio or even the expressions alone, only the parameter κ. In order to be able to estimate λ_{T_b} and χ_T we would need further data and alternative statistics. We can however estimate κ with the same method as above, using the ratio of λ_{up} and λ_{down} from the experimental data and using numerical integration over a range of κ values to find the value that minimises the error.

The ranges of parameter values used for the integration in both cases here could be informed by the typical values observed in Bearon & Durham (2019), i.e. $\chi \in [0, 1]$, $\kappa \in [0, 10]$.

As an example, simulations of the Twiddle Only model were used to estimate κ from the data, where $\lambda_{T_b} = 0.6, \chi_R = 1, \kappa = 0.5, dt = 0.1, \text{Ncells} = 200, t_N = 1000$, the simulations gave a value for $\lambda_{up}/\lambda_{down} = 0.4011/0.8972 = 0.4471$. Upon using these values in the analytic expressions, κ was estimated as 0.6330.

Results from the IBM

It was established above that at steady state, if cells can be modelled as set out by the model in Bearon & Durham (2019), then we should see either a uniform number of cells turning towards and away from an angle θ at steady state when

cells only reverse, or a number predicted by the von Mises distribution when they only twiddle. More specifically we should see the same number of cells travelling up the gradient and reversing as cells travelling down the gradient and reversing, though this will not be the case for twiddles. The automated data provided on reversals and twiddles was studied in relation to this result to see if it was observed experimentally.

It was possible to conduct a hypothesis test to investigate the number of turns after swimming up or down in the given data. This was tested using an exact Binomial test (R `binom.test`, Statistics Package, (R Core Team, 2020)), testing whether the probability of success (i.e. probability of turning when travelling down the gradient) was equal to 0.5. This would mean that cells travelling up the gradient are just as likely to turn as cells travelling down the gradient.

In order to prepare the data for the test, entries into twiddles and reversals were sorted into either up or down. Reversal data contained angles measured between $-\pi$ and π, thus meaning that angles in $(0, \pi)$ were classed as upward entries and those in $[-\pi, 0]$ were classed as downward. Twiddle data contained angles measured between 0 and 2π meaning that angles in $(0, \pi)$ were classed as upward entries and those in $[\pi, 2\pi]$ were classed as downward.

The results of the tests can be seen in table 4.8, where only the two-tailed alternative has been recorded as this is sufficient for our purpose here.

We see from these tests that when looking at reversals we see what we expect to at steady state - that the numbers of reversals coming from cells travelling both up and down the gradient are not significantly different. For twiddles the opposite is true, though this is also expected due to the exit bias meaning that the number of turns will not be uniform. Aside from in Experiment 1, we see significantly different numbers of twiddles when cells are travelling up and down the gradient. This confirms the key difference between reversals and twiddles in their steady state behaviour. These conclusions suggest that both twiddles and reversals could be modelled using Poisson processes, as long as consideration is given to the differences in the descriptions of their turn kernels and what happens with direction distributions at steady state.

Experiment	U vs D	**p-value**	**Conclusion**
Reversals			
Experiment 1	37 vs 45	0.4397	U = D
Experiment 2	37 vs 29	0.3891	U = D
Experiment 3	38 vs 32	0.5504	U = D
Pooled	112 vs 108	0.8398	U = D
Twiddles			
Experiment 1	38 vs 49	0.2836	U = D
Experiment 2	71 vs 101	0.0267	U $\neq$ D
Experiment 3	33 vs 56	0.0019	U $\neq$ D
Pooled	142 vs 206	7.1e-04	U $\neq$ D

Table 4.8: Results of the binomial tests carried out to investigate whether cells are just as likely to turn when travelling up the gradient as they are when travelling down the gradient. U = number of turns when travelling up the gradient, D = number of turns when travelling down the gradient. In all cases the significance level of 0.05 has been used to decide whether to reject the null hypothesis.

Ultimately our aim was to parametrize the IBM and automated data was intended to be used to obtain realistic parameter values in the way outlined above. When we came to use the experimental data to obtain these estimates and try to replicate what was seen in experiments, we found that obtaining the parameter estimates from the given data was not possible. This is due to the given data only containing records of reorientation events and not documenting the complete tracks of all cells spanning the whole experiment. This means that we cannot accurately estimate λ^D_{up} or λ^D_{down}, which require knowledge of the total number of cells moving in each direction and not just the ones that turn at any given time.

It was also impossible to estimate some important twiddle parameters with these quantities, suggesting that further data and summary statistics are needed. We also need some indication about all cells and whether they were swimming up or down the gradient, or to observe the distribution of times between turns, monitoring a few cells for a very long time or lots of cells for a shorter time.

4.3 Discussion and conclusions

This chapter has focused on exploring the modelling of the newly-discovered 'twiddling' mechanism observed in *P. aeruginosa* carrying out chemotaxis on a surface.

Before attempting to put a mathematical modelling framework in place in the form of an IBM, testing of three key hypotheses concerning entries into and exits from these twiddles and the occurrence of simultaneous reversals, was carried out. These hypotheses were tested on manually tracked, and in one case automatically tracked, data from three experimental replicates and both in the presence and absence of a chemoattractant gradient. The hypotheses themselves were proposed to answer three key questions about twiddles, answers to which were deemed necessary before an IBM could be built to try and model the observed behaviours. These were "Are exits from twiddles equally likely to be correct or not correct?", "Are exits from twiddles independent from entry into them?" and "Is exit from a twiddle independent of reversal within a twiddle, regardless of entry into the twiddle?".

The results from testing these hypotheses varied between manual and automated data sets. For hypothesis 1a which compares exits in the presence of a gradient and then in the absence of a gradient, the manual data concludes that in the Pooled case, there is no bias for correct exits over not correct in a gradient, but this bias is observed in the absence of a gradient. There was overwhelming evidence from the automated data analyses that there is bias for correct exits in a gradient when comparing the proportion of correct exits to not correct.

Hypothesis 1b compared the proportion of correct exits in both the presence and absence of a gradient. The automated data provided strong evidence for bias in correct exits in a gradient when compared to the absence of a gradient in all experimental cases. This result was also observed in the Pooled case in the manual data set. This leads us to believe that twiddles do indeed favourably orient bacteria in a direction which is up a chemoattractant gradient, more so than would be expected in an environment where this gradient is not present.

Hypothesis 2 tested independence between means of entry to, and exit from twiddles. It was shown that independence between entry and exit could be assumed in all cases, across all experiments and both in the presence and absence of a gradient.

The last of the hypotheses, hypothesis 3, investigated whether there was independence between twiddle exit direction and simultaneous reversal after a correct or not correct entry. A clear dependence between reversal and exit direction when entry into the twiddle had been correct was demonstrated, suggesting that the reversal steers a correctly entered twiddle that may result in travelling in the wrong direction, back on track up the gradient.

In summary, it seems that there is a bias for correct exits from twiddles and this bias is more profound in a gradient than in the absence of one. We can reasonably assume independence between entry and exit from twiddles, and exit from twiddles seems to be dependent on simultaneous reversal more so when bacteria are in a gradient than when they are not.

Based on this information, the intention was to build an IBM to model twiddles and reversals as observed in experiments, using the experimental data provided. The IBM proposed was based on the continuum model for chemotaxis by Bearon & Durham (2019), taking turn rates and parameter definitions from this work and using them on the individual level. The proposed methods for extracting realistic parameter values for this model from the experimental data were derived and explained, where possible, and the simulation procedures outlined for IBMs that look at reversals and twiddles separately. This was to ensure independence between orientation events, allowing Poisson processes to be implemented.

Making use of the steady direction distribution $f_E^R(\theta)$ it was shown that at steady state, when cells reverse, we should see the same numbers of cells turning towards an angle θ as away from it. When cells twiddle, $f_E^T(\theta)$ shows that we should see some bias in the number of cells turning away from angle θ and that this bias can be dictated by the von Mises distribution. To investigate whether these results were seen in the experimental data, hypothesis tests were carried

out on all experimental setups to check the numbers of turns in each case.

For reversals it was found that the number of cells reversing after swimming up the gradient was not significantly different from the number of cells reversing after swimming down the gradient. For twiddles this was not the case and in 3 of the 4 cases tested there was a significant difference in these numbers, suggesting that there is indeed bias. In each experimental case there were more cells twiddling after swimming down, suggesting that this bias is towards the upwards direction, possibly centred around $\pi/2$ as predicted. This draws out a key difference between reversals and twiddles and how they cause cells to behave at steady state.

This work has revealed some interesting results surrounding chemotaxis in surface-attached bacteria, though there are obvious caveats to these results given the reliability of the data used. The manually tracked data has the potential to be biased due to it being tracked by hand and there perhaps being an influence from some desired effect such as wanting twiddles to explain chemotactic bias in these cells. There is also the issue of the small sample sizes in this data set, especially in the testing of hypothesis 3 where in some cases there was only one or even no observations in a category, and this can affect the power of the tests used.

Overall a data-integrated approach to modelling twiddles occurring during chemotaxis has been taken, testing assumptions from experimental data and then using these results to inform subsequent IBMs. It has been shown that twiddles and reversals are different types of turn and have very different direction distributions at steady state, suggesting that care needs to be taken when modelling. The IBM has been informed by manual tracking data and has been successful in capturing the different types of turns observed experimentally in automated tracking data with larger sample sizes.

To further extend the model, twiddles and reversals could be modelled together, making allowances for the two different types of motion that could occur in each cell at each time step. This would involve defining parameters for both types of motion and accounting for twiddles that last several time steps as op-

posed to the instantaneous reversals. It would be interesting to look at the impact of including both types of tumble on chemotactic drift, as well as what effect varying the rates of each has.

The duration of twiddles is one of the novel elements of this type of motion, so it would be useful to study this in more detail and model the impact of different durations of twiddle on the resulting drift.

Bacterial populations are often large and to model them more accurately we could look at larger cell numbers in the simulations. This means consideration of population size is needed, for example introducing the effects of overcrowding on tumble rates and looking at cell-cell interactions. We would also benefit from studying correlations in orientation in larger populations, looking at orientation distributions over time and assessing the angles of consecutive tumbles. This could reveal more about the bias twiddles cause, and if there is in fact any memory between twiddles.

Finally, it would be interesting to consider how noise impacts the simulations. Using rotational diffusion to model noise in the simulations means that this can be adjusted as necessary, and with the other parameters in the model. The effects of varying noise, especially surrounding reversals could be studied further to examine how this affects population level parameters such as chemotactic drift velocity or mean squared displacement.

There is a need for further investigation into twiddles and reversals and how they could be modelled more realistically, along with collecting more data to be able to parametrize this model using experimental data, but the work presented here provides solid foundations upon which to build further, more realistic models of this kind.

Conclusions of the thesis

The work in this thesis has focused on data-integrated modelling of cell motility in 2 and 3 dimensions and using various stochastic models to study the behaviour of glioblastoma tumour cells and *P. aeruginosa* bacterial cells. In the first instance, a framework was developed based on the Persistent Random Walk model in 2 and 3 dimensions which parametrizes this model according to a specific data set, estimating cell speed S and persistence time P, and uses statistical measures to assess goodness-of-fit of the model to the data. This framework was tested on *in silico* data generated from the PRW model and then applied to experimental data sets from glioblastoma tumour spheroids grown *in vitro*.

When *in silico* data was used, estimates of motility parameters S and P were in good agreement with the known values and accompanying confidence intervals were narrow in both the 2- and 3-dimensional cases. Upon applying the framework to experimental data sets we found in both cases that the PRW model was insufficient to explain the observed data. In both cases this is likely due to the assumptions of the PRW not being satisfied, and it being too simple a model to capture the complexities of the motility.

In the 3D case it was found that S was unlikely to be a constant, and there were problems with having to cut short the data set used to estimate P due to the $VACF$ not being monotonic. Estimates of S were still reasonable compared to the experimental estimates, and P was estimated consistently across the three spheroids. In 2D, the sampling rate of the data was inadequate for the framework. This meant that P was very small, too small to be accurately estimated by the framework, and it became clear that correlation in velocity was unobservable over several time steps.

Thus we conclude from this work that the framework provides a good starting point for the rigorous analysis and testing of cell tracking data and relevant cell motility hypotheses, but it is essential that assumptions of the model being used are thoroughly checked before its application.

We have shown here plenty of examples of how to do this and added suggestions for how the model might be adapted to try and get better fits and estimates using the same workflow.

The first such example is to consider the geometry of the extracellular matrix that surrounds the migrating cells in vitro. There are many different examples of how to model this in the literature, given in the relevant chapters, and including such terms in the model could provide a better representation of migration in 3 dimensions, as well as highlighting the differences in migration when compared to 2 dimensions. This could also be enhanced by incorporating cell-cell interactions into the model, a natural extension given the proximity of cells to each other in a tumour microenvironment.

It would also be interesting to consider different versions of the MSD or $VACF$ as part of the model. As mentioned above, considering an MSD that represents sub- or superdiffusive could explain the underestimates seen when estimating MSD in our model. Adding in a sum of exponentials as the $VACF$, thus allowing persistence times to vary in the population, could see a better match between model predicted and experimental calculations of this correlation and therefore more accurate estimates of the persistence time parameter.

This naturally leads on to the idea of considering multiple subpopulations of cells in the model, a logical extension given the profoundness of heterogeneity among cancer cells. Allowing different values of parameters S and P in the model currently sees the framework give poor parameter estimates, but this problem may be solved by adjusting the expressions for MSD and $VACF$ accordingly.

In terms of further exploring and validating the model, we would require more data on the high-speed outlier cells that were identified by the framework. It would be interesting to explore this subset of cells further, perhaps also in terms of radial velocity, to see if they have different phenotypes from other cells which

weren't observed to move so fast. Studying the positions of these cells in the tumour could also highlight key differences in migration, for example, are these cells around the perimeter of the tumour and thus have more space to move, and are there correlations between these high speed movements, or between cells that may follow each other.

Finally, we could also explore the data given further by including more of it, finding a way to include cells that divide during the tracking and also investigating the impact of the frequency of sampling. More frequent sampling, particularly in 2D would allow us to see persistence over more time steps and hopefully produce more accurate parameter estimates as a result.

The framework can act as a diagnostic tool to reveal reasons why certain models may not provide the desired fit and can be altered, as discussed above, to include other biologically-informed terms. Changing the model in this way may then, for example, explain specific properties of cells, and some ways in which this could be approached have been discussed and demonstrated here to act as guidance for future work.

To complement the frequentist approach to statistics used in the framework, Bayesian methods were used to conduct parameter estimation and model selection. This work made use of the statistical measures from which parameters can be estimated, $RMSS$ and $\ln(VACF)$, fitting various regression models to these quantities and testing different priors for the parameters. The final analyses in this work produced parameter estimates that were comparable to the frequentist estimates, though had credibility intervals wider than confidence intervals produced in the frequentist analyses in most cases. Most notably, the experimental point estimates of $\hat{P}$ were consistently higher than those in the frequentist analyses, though estimates were still uncertain.

Model selection was then carried out to compare the AR(1) and AR(2) models assumed on the errors of the regression models that were fit to the $\ln(VACF)$ data. This was done using 3 methods, and overall the AR(1) model was preferred, though the strength of evidence for the AR(2) model in the *in silico* case with the largest sample size suggests that further consideration should be given to the

most appropriate choice of process to model the errors. This is also reiterated by the possibility of negative estimates of correlation obtained in some cases from both the AR(1) and AR(2) models suggested by wide credibility intervals which include 0.

There is much scope for further work on this project, particularly implementing more advanced Bayesian methods with the aim of elucidating more accurate parameter estimates. The use of ABC would allow us to directly use the SDE for the PRW model and estimate parameters using the summary statistics studied in this work, as well as other quantities considered relevant, for example mean speed or radial velocity. We could also employ RJMCMC to look at what may be the best correlation structure for the errors in the regression used in the current methodology.

This work has allowed comparison of the frequentist and Bayesian approaches to parameter estimation, highlighting benefits and drawbacks of both, but providing parameter estimates through a variety of different methods with varying degrees of accuracy. The Bayesian methods could be considered more subjective given the need to choose prior distributions for parameters, but in this case where we have relatively little knowledge of what the persistence time should be, being able to use a flat prior has been an advantage to the analysis. Given that this research is studying a problem in medicine, the opinions of clinicians and the scope for individual treatment could be incorporated into subsequent analysis in a Bayesian framework, and may be able to provide a path into individualized treatment in the future.

Overall, the analysis of the cell tracking data, both *in silico* and experimental, has demonstrated that rigorous testing of assumptions is essential for modelling the behaviour of cells, as well as integrating the data as much as possible into such models. The framework developed here provides a useful tool for being able to check these assumptions, both by looking at parameter estimates and by providing visual output of important statistical measures for the PRW model, but also as more general metrics for cell motility. The comparison of frequentist and Bayesian analyses of the same data sets has drawn out advantages and

disadvantages of both approaches, and perhaps suggests that using both methods is the best way to get as much information as possible from the data and use this to produce accurate and reliable parameter estimates.

The framework is intended as a tool for those who may want to conduct analyses of cell tracks but not necessarily have the mathematical background to carry out modelling work themselves. As such a simple model is used for initial analysis of the data, though modellers could change the analysis considerably if it is desired. The aim of the framework is to try and start to fill the tools gap for the analysis of, in particular, 3-dimensional cell tracking data which is becoming more common as technology advances. There is still further work to be done on a project such as this and many avenues for further exploration have been outlined in the relevant chapters.

The final chapter in this thesis focused on modelling bacterial chemotaxis in *P. aeruginosa* using an individual-based model based on a velocity jump process. Before creating this IBM, hypotheses created to probe model assumptions were tested to further study the potential bias of twiddles.

We found by testing these hypotheses that there does seem to be some bias for exits from twiddles up the gradient when such a gradient is present, and that entries to and exits from these twiddles are independent. Finally, twiddles that occur simultaneously with reversals seem to be exited from correctly more often when entry into them has been correct, suggesting that there is a relationship between twiddles and reversals that needs further exploration.

The IBM for simulating twiddles and reversals was outlined and demonstrated, including visualisations of cell tracks with and without the chemotactic bias, and plots of the orientation distributions over the simulation and the number of turns away from an angle once the simulation was supposed to have reached steady state. These plots confirmed that the simulations and the continuum model were in good agreement when it came to modelling reversals and twiddles. A hypothesis test comparing the numbers of turns of cells travelling up or down the gradient at steady state confirmed that these numbers were not significantly different in experimentally observed reversals, but were in most of

the cases of experimentally observed twiddles. This confirms, as far as we are able to investigate, that the continuum model predicts the behaviour of the experimentally observed cells, in that twiddles and reversals behave differently at steady state.

The IBM was unable to be parametrized with realistic values due to insufficient data, and recommendations for what would be needed to further study this were given in the chapter. In particular, there is a need for reliable paired data concerning twiddles. Data of this type would allow further study of the parameters outlined in this work, more powerful results from hypothesis tests where only manual data was used, and also further exploration of twiddles. We would for example be able to assess correlation distributions in turn angles, allowing further investigation of the bias individual twiddles ay cause as well as the overall impact on chemotactic drift.

Once appropriate data was provided there would be plenty of opportunity for further work on this project, including realistic parametrization of the IBM, and creating a model that reflects the dynamics of cells that are both reversing and twiddling, allowing further consideration of the effect they can have on each other as well as individually. Quantities outlined in this work could then be validated, such as the ratio $\lambda_{up}/\lambda_{down}$, as well as the chemotactic drift and bias in tumbles.

One of the main, overarching conclusions that can be made from the work in this thesis is that for experimentalists and modellers to work successfully when modelling cell motility, it is key that work is conducted in an iterative manner. Initial *in silico* simulations of cells moving in the manner in which is to be studied is a good starting point for being able to test model assumptions and whether the simulation is an accurate reflection of what is seen in experiments. Models can then be applied to initial experimental data and estimates calculated, and what follows should be back and forth between experimentalists and modellers once caveats are identified, be that with the data or the model. In this way, the work of both parties can be improved by insights gained from both *in silico* and *in vitro* or *in vivo* work. It is vital however to let the data guide the way and

be open to changing models or experiments where necessary, the importance of which is demonstrated throughout this work.

Bibliography

Adler, J. 1966 Chemotaxis in Bacteria. *Science* **153** (3737), 708–716.

Adler, J. 1969 Chemoreceptors in Bacteria. *Science* **166** (3913), 1588–1597.

Agosti, A., Giverso, C., Faggiano, E., Stamm, A. & Ciarletta, P. 2018 A personalized mathematical tool for neuro-oncology: A clinical case study. *International Journal of Non-Linear Mechanics* **107**, 170–181.

Akaike, H. 1973 Information Theory and an Extension of the Maximum Likelihood Principle. In *Proceedings of the Second International Symposium on Information Theory* (ed. B. N. Petrov & F. Csaki). Budapest: Akademiai Kiado.

Alirezaeizanjani, Z., Grossman, R., Pfeifer, V., Hintsche, M. & Beta, C. 2020 Chemotaxis strategies of bacteria with multiple run modes. *Science Advances* **6** (22), eaaz6153.

Alt, W. 1980 Biased Random Walk Models for Chemotaxis and Related Diffusion Approximations. *Journal of Mathematical Biology* **9**, 147–177.

Altindal, T., Xie, L. & Wu, X. 2011 Implications of Three-Step Swimming Patterns in Bacterial Chemotaxis. *Biophysical Journal* **100**, 32–41.

Anderson, A. R. A. & Quaranta, V. 2008 Integrative Mathematical Oncology. *Nature Reviews Cancer* **8**, 227–234.

Antoni, D., Burckel, H., Josset, E. & Noel, G. 2015 Three-Dimensional Cell Culture: A Breakthrough *in Vivo*. *International Journal of Molecular Sciences* **16**, 5517–5527.

Antonopoulos, M., Dionysiou, D., Stamatakos, G. & Uzunoglu, N. 2019 Three-dimensional tumor growth in time-varying chemical fields: a modeling framework and theoretical study. *BMC Bioinformatics* **20** (1), 442.

Antonopoulos, M. & Stamatakos, G. 2015 In Silico Neuro-Oncology: Brownian Motion-Based Mathematical Treatment as a Potential Platform for Modeling the Infiltration of Glioma Cells into Normal Brain Tissue. *Cancer Informatics* **14** (Supplement 4), 33–40.

Armitage, J. P.and Pitta, T. P., Viegant, M. A. S., Packer, H. L. & Ford, R. M. 1999 Transformations in Flagellar Structure of Rhodobacter sphaeroides and Possible Relationship to Changes in Swimming Speed. *Journal of Bacteriology* **181** (6), 4825–4833.

Arumugam, G. & Tyagi, J. 2021 Keller-Segel Chemotaxis Models: A Review. *Acta Applicandae Mathematicae* **171** (6).

Ashby, D. 2006 Bayesian statistics in medicine: A 25 year review. *Statisitcs in Medicine* **25** (21), 3589–3631.

Baele, G., Lemey, P., Rambaut, A. & Suchard, M. A. 2017 Adaptive MCMC in Bayesian phylogenetics: an application to analyzing partitioned data in BEAST. *Bioinformatics* **33** (12), 1798–1805.

Bassetti, M., Vena, A. & Croxatto, A. et al 2018 How to manage Pseudomonas aeruginosa infections. *Drugs in Context* **7**, 212527.

Bayes, T. 1763 LII. An essay towards solving a problem in the doctrine of chances. By the late Rev. Mr. Bayes, F. R. S. communicated by Mr. Price, in a letter to John Canton, A. M. F. R. S. *Philosophical Transactions of the Royal Society of London* **53**, 370–418.

Bearon, R. N. & Durham, W. M. 2019 A model of strongly biased chemotaxis reveals the trade-offs of different bacterial migration strategies. *Mathematical Medicine and Biology: A Journal of the IMA* **37**, 83–116.

Beaumont, M. A., Zhang, W. & Balding, D. J. 2002 Approximate Bayesian Computation in Population Genetics. *Genetics* **162** (4), 2025–2035.

Berg, H. C. & Brown, D. A. 1972 Chemotaxis in Escherichia coli analysed by Three-dimensional Tracking. *Nature* **239**, 500–504.

Berger, J. 2006 The Case for Objective Bayesian Analysis. *Bayesian Analysis* **1** (3), 385–402.

Berger, J. O. & Bernardo, J. M. 1992 On the development of reference priors. In *Bayesian Statistics 4* (ed. J. M. Bernardo, J. O. Berger, A. P. Dawid & A. F. M. Smith), pp. 35–60. Oxford University Press.

Bernardo, J. M. 1979 Reference Posterior Distributions for Bayesian Inference. *Journal of the Royal Statistical Society. Series B (Methodological)* **41** (2), 113–147.

Bois, F. Y., Hsieh, N., Gao, W., CHiu, W. A. & Reisfeld, B. 2020 Well-Tempered MCMC simulations for population pharmacokinetic models. *Journal of pharmacokinetics and pharmacodynamics* **47** (6), 543–559.

Brockwell, P. J. & Davis, R. A. 2016 *Introduction to Time Series and Forecasting*, 3rd edn. Springer, Cham.

Brown, D. A. & Berg, H. C. 1974 Temporal Stimulation of Chemotaxis in Escherichia coli. *PNAS* **71** (4), 1388–1392.

Buensuceso, R. N. C., Daniel-Ivad, M., Kilmury, S. L. N., Leighton, T. L., Harvey, H., Howell, P. L. & Burrows, L. L. 2017 Cyclic AMP-Independent Control of Twitching Motility in Pseudomonas aeruginosa. *Journal of Bacteriology* **199**, e00188–17.

Burr, T. & Skurikhin, A. 2013 Selecting Summary Statistics in Approximate Bayesian Computation for Calibrating Stochastic Models. *BioMed Research International* **2013**, 210646.

Calvez, V., Raoul, G. & Schmeiser, C. 2015 Confinement by biased velocity jumps: Aggregation of escherichia coli. *Kinetic and Related Models* **8** (4), 651–666.

Campos, D., Méndez, V. & Llopis, I. 2010 Persistent random motion: Uncovering cell migration dynamics. *Journal of Theoretical Biology* **267** (4), 526–534.

Cancer Research UK 2021 Cancer Statistics for the UK. https://www.cancerresearchuk.org/health-professional/cancer-statistics-for-the-uk, accessed: 03/05/2021.

Cauchemez, S., Carrat, F., Viboud, C., Valleron, A. J. & Y., Boëlle. P. 2004 A Bayesian MCMC approach to study transmission of influenza: application to household longitudinal data. *Statistics in Medicine* **23** (22), 3469–3487.

Cherstvy, A. G., Nagel, O., Beta, C. & Metzler, R. 2018 Non-Gaussianity, population heterogeneity, and transient superdiffusion in the spreading dynamics of amoeboid cells. *Physical Chemistry Chemical Physics* **20** (35), 23034–23054.

Codling, E. A., Plank, M. J. & Benhamou, S. 2008 Random walk models in biology. *Journal of the Royal Society Interface* **5**, 813–834.

Collis, J., Connor, A. J., Paczkowski, M., Kannan, P. & Pitt-Francis, J. et al 2017 Bayesian Calibration, Validation and Uncertainty Quantification for Predictive Modelling of Tumour Growth: A Tutorial. *Bulletin of Mathematical Biology* **79**, 939–974.

Colombo, M. C., Giverso, C., Faggiano, E., Boffano, C., Acerbi, F. & Ciarletta, P. 2015 Towards the Personalized Treatment of Glioblastoma: Integrating Patient-Specific Clinical Data in a Continuous Mechanical Model. *PLoS ONE* **10** (7), e0132887.

da Costa, J. M. J., Orlande, H. R. B. & da Silva, W. B. 2018 Model selection and parameter estimation in tumor growth models using approximate Bayesian computation-ABC. *Computational and Applied Mathematics* **37**, 2795–2815.

Cziròk, A., Vicsek, M. & Vicsek, T. 1999 Collective motion of organisms in three dimensions. *Physica A: Statistical Mechanics and its Applications* **264**, 299–304.

Deisboeck, T. S., Zhang, L., Yoon, J. & Costa, J. 2009 In silico cancer modeling: Is it ready for prime time? *Nature Clinical Practice Oncology* **6**, 34–42.

Del Moral, P., Doucet, A. & Jasra, A. 2006 Sequential Monte Carlo Samplers. *Journal of the Royal Statistical Society. Series B (Methodological)* **68** (3), 411–436.

Del Moral, P., Doucet, A. & Jasra, A. 2012 An adaptive sequential Monte Carlo method for approximate Bayesian computation. *Statistics and Computing* **22**, 1009–1020.

Dempster, A. P., Laird, N. M. & Rubin, D. B. 1977 Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)* **39** (1), 1–38.

Denwood, Matthew J. 2016 runjags: An R package providing interface utilities, model templates, parallel computing methods and additional distributions for MCMC models in JAGS. *Journal of Statistical Software* **71** (9), 1–25.

Dickey, J. M. & Lientz, B. P. 1970 The Weighted Likelihood Ratio, Sharp Hypotheses about Chances, the Order of a Markov Chain. *Annals of Mathematical Statistics* **41** (1), 214–226.

Dieterich, P., Klages, R., Preuss, R. & Schwab, A. 2008 Anomalous dynamics of cell migration. *PNAS* **105** (2), 459–463.

Diggle, P. J. & Gratton, R. J. 1984 Monte Carlo Methods of Inference for Implicit Statistical Models. *Journal of the Royal Statistical Society. Series B (Methodological)* **46** (2), 193–227.

Dimilla, P. A., Quinn, J. A., Albelda, S. M. & Lauffenburger, D. A. 1992 Measurement of individual cell migration parameters for human tissue cells. *AIChE Journal* **38** (7), 1902–1104.

Driscoll, M. K. & Danuser, G. 2015 Quantifying modes of 3D cell migration. *Trends in Cell Biology* **25** (12), 749–759.

Duane, S., D., Kennedy A., Pendleton, B. J. & Roweth, D. 1987 Hybrid Monte Carlo. *Physics Letters B* **195** (2), 216–222.

Dunn, G. A. & Brown, A. F. 1987 A Unified Approach to Analysing Cell Motility. *Journal of Cell Science. Supplement* **8**, 81–102.

Ellis, H. P., Greenslade, M., Powell, B., Spiteri, I., Sottoriva, A. & Kurian, K. M. 2015 Current Challenges in Glioblastoma: Intratumour Heterogeneity, Residual Disease, and Models to Predict Disease Recurrence. *Frontiers in Oncology* **5** (251).

Erban, R. & Othmer, H. G. 2004 From Individual to Collective Behaviour in Bacterial Chemotaxis. *SIAM Journal on Applied Mathematics* **65** (2), 361–391.

Erban, R. & Othmer, H. G. 2005 From Signal Transduction to Spatial Pattern Formation in E. coli: A Paradigm for Multiscale Modeling in Biology. *Multiscale Modeling and Simulation* **3** (2), 362–394.

Erban, R. & Othmer, H. G. 2007 Taxis equations for amoeboid cells. *Journal of Mathematical Biology* **54**, 847–885.

Ferrer, J., Prats, C. & Lopez, D. 2008 Individual-based Modelling: An Essential Tool for Microbiology. *Journal of Biological Physics* **34**, 19–37.

Fraley, S. I., Wu, P., He, L., Feng, Y., Krisnamurthy, R., Longmore, G. D. & Wirtz, D. 2015 Three-dimensional matrix fiber alignment modulates cell migration and MT1-MMP utility by spatially and temporally directing protrusions. *Scientific Reports* **5**, 14580.

Friedl, P., Sahai, E., Weiss, S. & Yamada, K. M. 2012 New dimensions in cell migration. *Nature Reviews Molecular Cell Biology* **13**, 743–747.

Gail, M. & Boone, C. 1970 The Locomotion of Mouse Fibroblasts in Tissue Culture. *Biophysical Journal* **10**, 980–993.

Gardiner, C. W. 2009 *Stochastic Methods: A Handbook for the Natural and Social Sciences*, 4th edn., *Springer Series in Synergetics*, vol. 13. Springer-Verlag Berlin Heidelberg.

Gelfand, A. E. & Smith, A. F. M. 1990 Sampling-Based Approaches to Calculating Marginal Densities. *Journal of the American Statistical Association* **85** (410), 398–409.

Gelman, A., Carlin, J.B., Stern, H.S., Dunson, D.B., Vehtari, A. & Rubin, D.B. 2013 *Bayesian Data Analysis, Third Edition*. Taylor & Francis.

Gelman, A., Hwang, J. & Vehtari, A. 2013 Understanding predictive information criteria for Bayesian models. *Statistics and Computing* **24**, 997–1016.

Gelman, A. & Rubin, D. B. 1992 Inference from Iterative Simulation Using Multiple Sequences. *Statistical Science* **7** (4), 457–511.

Geman, S. & Geman, D. 1984 Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **PAMI-6** (6), 721 – 741.

Gerlee, P. & Nelander, S. 2012 The Impact of Phenotypic Switching on Glioblastoma Growth and Invasion. *PLoS Computational Biology* **8** (6), e1002556.

Gilks, W. R., Thomas, A. & Spiegelhalter, D. J. 1994 A language and program for complex Bayesian modelling. *The Statistician* **43** (1), 169–177.

Ginovart, M., Lòpez, D. & Valls, J. 2002 INDISIM, An Individual-based Discrete Simulation Model to Study Bacterial Cultures. *Journal of Theoratical Biology* **214**, 305–319.

Good, I. J. 1979 Studies in the History of Probability and Statistics. XXXVII A. M. Turing's Statistical Work in World War II. *Biometrika* **66** (2), 393–396.

Good, I. J. 1985 Weight of Evidence: A Brief Survey. In *Bayesian Statistics 2* (ed. J. Bernardo, M. DeGroot, D. Lindley & A. Smith), pp. 249–269. North Holland.

Green, P. J. 1995 Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika* **82** (4), 711–732.

Green, P. L. & Maskell, S. 2016 Parameter estimation from big data using a sequential monte carlo sampler. *27th ISMA Conference on Noise and Vibration Engineering, Leuven, Belgium, September 19–21* .

Grimm, V. 1999 Ten years of individual-based modelling in ecology: what have we learned and what could we learn in the future? *Ecological Modelling* **115** (2-3), 129–148.

Grimm, V., Berger, U. & Bastiansena, F. et al 2006 A standard protocol for describing individual-based and agent-based models. *Ecological Modelling* **198** (1-2), 115–126.

Hakkinen, K. M., Harunaga, J. S., Doyle, A. D. & Yamada, K. M. 2011 Direct comparisons of the morphology, migration, cell adhesions, and actin cytoskeleton of fibroblasts in four different three-dimensional extracellular matrices. *Tissue Engineering: Part A* **17** (5-6), 713–724.

Hamis, S., Powathil, G. G. & Chaplain, M. A. J. 2019 Blackboard to Bedside: A Mathematical Modeling Bottom-Up Approach Toward Personalized Cancer Treatments. *JCO Clinical Cancer Informatics* **3**, 1–11.

Harms, R. L. & Roebroeck, A. 2018 Robust and Fast Markov CHain Monte Carlo Sampling of Diffusion MRI Microstructure Models. *Frontiers in Neuroinformatics* **12** (97).

Harrison, J. U. & Baker, R. E. 2018 The impact of temporal sampling resolution on parameter inference for biological transport models. *PLOS Computational Biology* **14** (6), e1006235.

Hastings, W. K. 1970 Monte Carlo sampling methods using Markov chains and their applications. *Biometrika* **57** (1), 97–109.

Hathout, L., Patel, V. & Wen, P. 2016 A 3-dimensional DTI MRI-based model of GBM growth and response to radiation therapy. *International Journal of Oncology* **49** (3), 1081–1087.

Hawkins-Daarud, A., Prudhomme, S., van der Zee, K. G. & Tinsley Oden, J. 2013 Bayesian calibration, validation, and uncertainty quantification of diffuse interface models of tumor growth. *Journal of Mathematical Biology* **67**, 1457–1485.

Hellweger, F. L. & Bucci, V. 2009 Individual-based Modelling: An Essential Tool for Microbiology. *Ecological Modelling* **220** (1), 8–22.

Hellweger, F. L., Clegg, R. J., Clark, J. R., Plugge, C. M. & Kreft, J. 2016 Advancing microbial sciences by individual-based modelling. *Nature Reviews Microbiology* **14**, 461–471.

Hillen, T. & Painter, K. J. 2009 A user's guide to PDE models for chemotaxis. *Journal of Mathematical Biology* **58**, 183–217.

Hinton, G. E. & van Camp, D. 1993 Keeping the neural networks simple by minimizing the description length of the weights. In *COLT '93: Proceedings of the sixth annual conference on Computational learning theory* (ed. L. Pitt), pp. 5–13. Association for Computing Machinery, New York, NY, United States.

Hintsche, M., Waljor, V., Grossman, R., Kühn, M. J., Thormann, K. M., Peruani, F. & Beta, C. 2017 A polar bundle of flagella can drive

bacterial swimming by pushing, pulling, or coiling around the cell body. *Scientific Reports* **7** (16771).

Hoarau-Véchot, J., Rafii, A., Touboul, C. & Pasquier, J. 2018 Halfway between 2D and animal models: Are 3D cultures the ideal tool to study cancer-microenvironment interactions? *International Journal of Molecular Sciences* **19** (181).

Huerta, G. 2012 Lecture notes STAT 574 Introduction to Statistical Theory.

Husmeier, D. & McGuire, G. 2002 Detecting recombination with MCMC. *Bioinformatics* **18** (suppl1), S345–S353.

IBM Corp 2017 *IBM SPSS Statistics for Windows, Version 25.0*. IBM Corp, Armonk, NY.

Ihsani, A., Sitek, A., Petibon, Y., Ma, C., Han, P. El Fakhri, G. & Ouyang, J. 2018 Markov Chain Monte Carlo Estimation of Non-stationary PET Kinetic Parameters Compartment Models: A Flow Phantom Study. *Journal of Nuclear Medicine* **59** (supplement 1), 1721.

Jackson, P. R., Juliano, J., Hawkins-Daarud, A., Rockne, R. C. & Swanson, K. R. 2015 Patient-Specific Mathematical Neuro-Oncology: Using a Simple Proliferation and Invasion Tumor Model to Inform Clinical Practice. *Bulletin of Mathematical Biology* **77** (5), 846–856.

Jarosz, A. F. & Wiley, J. 2014 What Are the Odds? A Practical Guide to Computing and Reporting Bayes Factors. *Journal of Problem Solving* **7** (1), Article 2.

Jeffreys, H. 1939 *Theory of Probability*, 1st edn. The Clarendon Press, Oxford.

Jeffreys, H. 1946 An invariant form for the prior probability in estimation problems. *Proceedings of the Royal Society of London A. Mathematical, Physical and Engineering Sciences* **186**, 453–461.

Kass, R. E. & Rafferty, A. E. 1995 Bayes Factors. *Journal of the American Statistical Society* **90** (430), 773–795.

Keller, E. F. & Segel, L. A. 1971 Model for Chemotaxis. *Journal of Theoretical Biology* **30** (2), 225–234.

Kooperberg, Charles 2020 *logspline: Routines for Logspline Density Estimation*. R package version 2.1.16.

Kreft, J.and Picioreanu, C., Wimpenny, J. W. T. & van Loosdrecht, M. C. M. 2001 Individual-based modelling of biofilms. *Microbiology* **147** (11), 2897–2912.

Kreft, J., Booth, G. & Wimpenny, J. W. T. 1998 BacSim, a simulator for individual-based modelling of bacterial colony growth. *Microbiology* **144** (Pt 12), 3275–3287.

Kreft, J., Plugge, C. M., Prats, C., Leveau, J. H. J., Zhang, W. & Hellweger, F. L. 2017 From Genes to Ecosystems in Microbiology: Modeling Approaches and the Importance of Individuality. *Frontiers in Microbiology* **8**, 2299.

Kursawe, J., Baker, R. E. & Fletcher, A. G. 2018 Approximate Bayesian computation reveals the importance of repeated measurements for parameterising cell-based models of growing tissues. *Journal of Theoretical Biology* **14** (443), 66–81.

Laplace, P. 1812 Théorie analytique des probabilités. *Courcier* .

Lardon, L. A., Merkey, B. V., Martins, S., Dötsch, A., Picioreanu, C., Kreft, J. & Smeths, B. F. 2011 iDynoMiCS: next-generation individual-based modelling of biofilms. *Environmental Microbiology* **13** (9), 2416–2434.

Lauffenburger, D. A. & Horwitz, A. F. 1996 Cell Migration: A Physically Integrated Molecular Process. *Cell* **84** (3), 359–369.

Lauga, E. 2016 Bacterial Hydrodynamics. *Annual Reviews of Fluid Mechanics* **48**, 105–130.

Lê, M., Delingette, H. & Kalpathy-Cramer, J. et al 2015 Bayesian Personalization of Brain Tumour Growth Model. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015* (ed. Navab N., Hornegger J., Wells W. & Frangi A.), *Lecture Notes in Computer Science*, vol. 9350. Springer, Cham.

Lee, B., Zhou, X., Riching, K., Eliceiri, K. W., Keely, P. J., Guelcher, S. A., Weaver A. M. & Jiang, Y. 2014 A Three-Dimensional Computational Model of Collagen Network Mechanics. *PLoS One* **9** (11), e111896.

Lee, J. 2018 Insights into cell motility provided by the iterative use of mathematical modeling and experimentation. *AIMS Biophysics* **5**, 97–124.

Li, Q., Fan, X., Liang, T. & Li, S 2011 An MCMC algorithm for detecting short adjacent repeats shared by multiple sequences. *Bioinformatics* **27** (13), 1772–1779.

Lipková, J., Angelikopoulos, P., Wu, S. & Alberts, E. et al 2019 Personalized Radiotherapy Design for Glioblastoma: Integrating Mathematical Tumor Models, Multimodal Scans, and Bayesian Inference. *IEEE Transactions on Medical Imaging* **38** (8), 1875–1884.

Liu, J. S. & Chen, R. 1998 Sequential Monte Carlo Methods for Dynamic Systems. *Journal of the American Statistical Association* **93** (443), 1032–1044.

Liu, Z. 2010 Consensus of the 3-dimensional Vicsek model. In *Proceedings of the 29th Chinese Control Conference, Bejing, China*, pp. 4635–4640. IEEE.

Loosley, A. J., O'Brien, X. M., Reichner, J. S. & Tang, J. X. 2015 Describing Directional Cell Migration with a Characteristic Directionality Time. *PLoS ONE* **10** (5), e0127425.

Lowengrub, J. S., Frieboes, H. B., Jin, F., Chuang, Y. L., Li, X., MacKlin, P., Wise, S. M. & Cristini, V. 2010 Nonlinear modelling of cancer: bridging the gap between cells and tumours. *Nonlinearity* **23** (1), R1–R91.

Luzhansky, I. D., Schwartz, A. D., Cohen, J. D., MacMunn, J. P., Barney, L. E., Jansen, L. E. & Peyton, S. R. 2018 Anomalously diffusing and persistently migrating cells in 2D and 3D culture environments. *APL Bioengineering* **2** (2), 026112.

Macklin, P., Edgerton, M. E., Lowengrub, J. S. & Cristini, V. 2010 Discrete cell modelling. In *Multiscale modeling of cancer: an integrated experimental and mathematical modeling approach* (ed. V. Cristini & J. Lowengrub), pp. 88–122. Cambridge University Press.

MacLehose, R. F., Dunson, D. B., Herring, A. H. & Hoppin, J. A. 2007 Bayesian Methods for Highly Correlated Exposure Data. *Epidemiology* **18** (2), 199–207.

Mathworks 2019 *MATLAB R2019a*. The Mathworks, Inc., Natick, Massachusetts.

Matsiaka, O. M., Baker, R. E., Shah, E. T. & Simpson, M. J. 2019 Mechanistic and experimental models of cell migration reveal the importance of cell-to-cell pushing in cell invasion. *Biomedical Physics and Engineering Express* **5**, 045009.

Matsutani, T., Ueno, Y., Fukunaga, T. & Hamada, M. 2019 Discovering novel mutation signatures by latent Dirichlet allocation with variational Bayes inference. *Bioinformatics* **35** (22), 4543–4552.

McDonald, J. H. 2014 *Handbook of Biological Statistics*, 3rd edn. Sparky House Publishing, Baltimore, Maryland.

Metropolis, N., Rosenbluth, A. W., N., Rosenbluth M. & H., Teller A. 1953 Equation of State Calculations by Fast Computing Machines. *The Journal of Chemical Physics* **21**, 1087–1092.

Metzner, C., Mark, C., Steinwachs, J., Lautscham, L., Stadler, F. & Fabry, B. 2015 Superstatistical analysis and modelling of heterogeneous random walks. *Nature Communications* **6**, 7516.

Mierke, C. T. 2015 Physical view on migration modes. *Cell Adhesion and Migration* **9** (5), 367–379.

Mitchison, T. J. & Cramer, L. P. 1996 Actin-Based Cell Motility and Cell Locomotion. *Cell* **84** (3), 371–379.

Neal, R. M. & Hinton, G. E. 1993 A New View of the EM Algorithm that Justifies Incremental and Other Variants. In *Learning in Graphical Models*, pp. 355–368. Kluwer Academic Publishers.

Neal, R. M. & Hinton, G. E. 1998 A View of the EM Algorithm that Justifies Incremental, Sparse, and other Variants. In *Learning in Graphical Models. Volume 89* (ed. M. I. Jordan). Springer, Dordrecht.

Ogundijo, O. E. & Wang, X. 2018 Characterization of tumor heterogeneity by latent haplotypes: a sequential Monte Carlo approach. *PeerJ* **30** (6), e4838.

Ogundijo, O. E., Zhu, K., Wang, X. & Anastassiou, D. 2019 A sequential Monte Carlo algorithm for inference of subclonal structure in cancer. *PLOS One* **14** (1), e0211213.

O'Hagan, A. 2004 Bayesian statistics: principles and benefits. In *Bayesian Statistics and Quality Modelling in the Agro-food Production Chain* (ed. M. A. J. VanBoekel, A. Stein & A. H. C. VanBruggen), *Wageningen UR Frontis Series*, vol. 3, pp. 31–45.

Oliveira, N. M., Foster, K. R. & Durham, W. M. 2016 Single-cell twitching chemotaxis in developing biofilms. *PNAS* **113** (23), 6532–6537.

Othmer, H. G., Dunbar, S. R. & Alt, W. 1988 Models of dispersal in biological systems. *Journal of Mathematical Biology* **26**, 263–298.

Othmer, H. G. & Xue, C. 2013 The Mathematical Analysis of Biological Aggregation and Dispersal: Progress, Problems and Perspectives. In *Dispersal, Individual Movement and Spatial Ecology. Lecture Notes in Mathematics, Volume 2071.* (ed. M. Lewis, P. Maini & S. Petrovskii), pp. 79–127. Springer, Berlin, Heidelberg.

Parisi, G. 1998 *Statistical Field Theory*. Avalon Publishing.

Parkhurst, M. R. & Saltzman, W. M. 1992 Quantification of human neutrophil motility in three-dimensional collagen gels - Effect of collagen concentration. *Biophysical Journal* **61**, 306–315.

Patlak, C. S. 1953 Random Walk with Persistence and External Bias. *Bulletin of Mathematical Biophysics* **15**, 311–338.

Paul, C. D., Hung, W., Wirtz, D. & Konstantopoulos, K. 2016 Engineered Models of Confined Cell Migration. *Annual Review of Biomedical Engineering* **18**, 159–180.

Paul, C. D., Mistriotis, P. & Konstantopoulos, K. 2017 Cancer cell motility: lessons from migration in confined spaces. *Nature Reviews Cancer* **17** (2), 131–140.

Peterson, C. & Anderson, J. R. 1987 A Mean Field Theory Learning Algorithm for Neural Networks. *Complex Systems* **1**, 995–1019.

Plaza, R. G. 2019 Derivation of a bacterial nutrient-taxis system with doubly degenerate cross-diffusion as the parabolic limit of a velocity-jump process. *Journal of Mathematical Biology* **78**, 1681–1711.

Plummer, M. 2003 JAGS: A program for analysis of Bayesian graphical models using Gibbs sampling .

Plummer, M. 2019 *rjags: Bayesian Graphical Models using MCMC*. R package version 4-10.

Pravitasari, A. A., Hermanto, Y. P. & Iriawan, N. et al 2019 MRI-based brain tumor segmentation using Gaussian mixture model with reversible jump Markov chain Monte Carlo algorithm. *AIP Conference Proceedings* **2194**, 020085.

R Core Team 2020 *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Rangarajan, R. & Zaman, M. H. 2008 Modeling cell migration in 3D: Status and challenges. *Cell Adhesion and Migration* **2** (2), 106–109.

Richards, R., Mason, D., Levy, R., Bearon, R. & See, V. 2018 4D imaging and analysis of multicellular tumour spheroid cell migration and invasion. *bioRxiv Preprint* .

Rockne, R. C., Hawkins-Daarud, A., Swanson, K. R., Sluka, J. P., Glazier, J. A., Macklin, P., Hormuth, D. A. & Jarrett, A. M. et al 2019 The 2019 mathematical oncology roadmap. *Physical Biology* **16**, 041005.

Rockne, R. C., Trister, A. D., Jacobs, J., Hawkins-Daarud, A. J., Neal, M. L., Hendrickson, K. Mrugala, M. M., Rockhill, J. K. & Kinahan, P. et al 2015 A patient-specific computational model of hypoxia-modulated radiation resistance in glioblastoma using ^{18}F-FMISO-PET. *Journal of the Royal Society Interface* **12** (103), 20141174.

Rosser, G., Baker, R. E., Armitage, J. P. & Fletcher, A. G. 2014 Modelling and analysis of bacterial tracks suggest an active reorientation mechanism in Rhodobacter sphaeroides. *Journal of the Royal Society Interface* **11**, 20140320.

Rosser, G., Fletcher, A. G., Wilkinson, D. A., de Beyer, J. A. & Yates, C. A., et al. 2013 Novel Methods for Analysing Bacterial Tracks Reveal Persistence in Rhodobacter sphaeroides. *PLOS Computational Biology* **9** (10), e1003276.

Rousset, M. & Samaey, G. 2013 Individual-Based Models for Bacterial Chemotaxis in the Diffusion Asymptotics. *Mathematical Models and Methods in Applied Sciences* **23** (11), 2005–2037.

Rubin, D. 1984 Bayesianly Justifiable and Relevant Frequency Calculations for the Applied Statisitcian. *The Annals of Statistics* **12** (4), 1151–1172.

Schälte, Y. & Hasenauer, J. 2020 Efficient exact inference for dynamical systems with noisy measurements using sequential approximate Bayesian computation. *Bioinformatics* **36** (Supplement 1), i551–i559.

Schlüter, D. K., Ramis-Conde, I. & Chaplain, M. A. J. 2012 Computational Modeling of Single-Cell Migration: The Leading Role of Extracellular Matrix Fibers. *Biophysical Journal* **103**, 1141–1151.

Scianna, M. & Preziosi, L. 2014 A cellular Potts model for the MMP-dependent and -independent cancer cell migration in matrix microtracks of different dimensions. *Computational Mechanics* **53** (3), 485–497.

Scott, M., Żychaluk, K. & Bearon, R. N. 2021 A mathematical framework for modelling 3D cell motility; applications to glioblastoma cell migration. *Mathematical Medicine and Biology* **38** (3), 333–354.

Selmeczi, D. & Mosler, S. et al 2005 Cell motility as persistent random motion: theories from experiments. *Biophysical Journal* **89**, 912–931.

Sepúlveda, N., Petitjean, L., Cochet, O., Grasland-Mongrain, E., Silberzan, P. & Hakim, V. 2013 Collective Cell Motion in an Epithelial Sheet Can Be Quantitatively Described by a Stochastic Interacting Particle Model. *PLoS Computational Biology* **9**, e1002944.

Spiegelhalter, D. J., Best, N. G., Carlin, B. P. & van der Linde, A. 2002 Bayesian measures of model complexity and fit. *Journal of the Royal Statistical Society. Series B (Methodological)* **64** (4), 583–639.

Spiegelhalter, D. J., Best, N. G., Carlin, B. P. & van der Linde, A. 2014 The deviance information criterion: 12 years on. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **76** (3), 485–493.

Stein, A. M., Vader, D. A., Sander, L. M. & Weitz, D. A. 2007 A Stochastic Model of Glioblastoma Invasion. In *Mathematical Modeling of Biological Systems, Vol I: Cellular Biophysics, Regulatory Networks, Development, Biomedicine, and Data Analysis* (ed. A. Deutsch, L. Brusch, H. Byrne, G. DeVries & H. Herzel), pp. 217–224. Birkhäuser Boston.

Stokes, C. L. & Lauffenburger, D. A. 1991 Migration of individual microvessel endothelial cells: stochastic model and parameter measurement. *Journal of Cell Science* **99** (Part 2), 419–430.

Strateva, T. & Yordanov, D. 2009 Pseudomonas aeruginosa – a phenomenon of bacterial resistance. *Journal of Medical Microbiology* **58**, 1133–1148.

Swanson, K. R., Rostomily, R. C. & Alvord, E. C. 2008 A mathematical modelling tool for predicting survival of individual patients following resection of glioblastoma: a proof of principle. *British Journal of Cancer* **98** (1), 113–119.

Takagi, H., Sato, M. J., Yanagida, T. & Ueda, M. 2008 Functional Analysis of Spontaneous Cell Movement under Different Physiological Conditions. *PLoS ONE* **3** (7), e2468.

Tavaré, S., Balding, D. J., Griffiths, R. C. & Donnelly, P. 1997 Inferring Coalescence Times from DNA Sequence Data. *Genetics* **145**, 505–518.

Taylor-King, J. P., van Loon, E. E., Rosser, G. & Chapman, S. J. 2015 From Birds to Bacteria: Generalised Velocity Jump Processes with Resting States. *Bulletin of Mathematical Biology* **77**, 1213–1236.

Toni, T, Welch, D., Strelkowa, N., Ipsen, A. & Stumpf, M. P. H. 2009 Approximate Bayesian computation scheme for parameter inference and model selection in dynamical systems. *Interface: A Journal of the Royal Society* **6**, 187–202.

Tozluoğlu, M., Tournier, A. L., Jenkins, R. P., Hooper, S., Bates, P. A. & Sahai, E. 2013 Matrix geometry determines optimal cancer cell migration strategy and modulates response to interventions. *Nature Cell Biology* **15**, 751–762.

Trägårdh, M., Chappell, M. J., Ahnmark, A., Lindén, D., Evans, N. D. & Gennemark, P. 2016 Input estimation for drug discovery using

optimal control and Markov chain Monte Carlo approaches. *Journal of Pharmacokinetics and Pharmacodynamics* **43**, 207–221.

Tranquillo, R. T. & Lauffenburger, D. A. 1987 Stochastic model of leukocyte chemosensory movement. *Journal of Mathematical Biology* **25** (3), 229–262.

Treloar, K., Simpson, M. & McCue, S. 2011 Velocity-jump models with crowding effects. *Physical Review E* **84** (6), 1–13.

Uhlenbeck, G.E. & Ornstein, L.S. 1930 On the theory of the Brownian motion. *Physical Review* **36** (5), 0823–0841.

Upadhyaya, A., Rieu, J. P., Glazier, J. A. & Sawada, Y. 2001 Anomalous diffusion and non-Gaussian velocity distribution of Hydra cells in cellular aggregates. *Physica A - Statistical Mechanics and its Applications* **293** (3-4), 549–558.

de Valpine, P., Turek, D., Paciorek, C. J., Anderson-Bergman, C., Temple Lang, D. & Bodik, R. 2017 Programming with models: writing statistical algorithms for general model structures with NIMBLE. *Journal of Computational and Graphical Statistics* **26**, 403–417.

van den Berg, R. G. 2021 Chi-Square Independence Test – What and Why? https://www.spss-tutorials.com/chi-square-independence-test/, accessed: 28/05/2021.

Vicsek, T., Cziròk, A., Ben-Jacob, E., Cohen, I. & Shochet, O. 1995 Novel Type of Phase Transition in a System of Self-Driven Particles. *Physical Review Letters* **75**, 1226–1229.

Wagenmakers, E., Lodewyckx, T., Kuriyal, H. & Grasman, R. 2010 Bayesian hypothesis testing for psychologists: A tutorial on the Savage–Dickey method. *Cognitive Psychology* **60**, 158–189.

Walsh, B. 2002 Lecture notes EEB596z.

Wang, C., Tong, X. & Yang, F. 2014 Bioengineered 3D Brain Tumor Model To Elucidate the Effects of Matrix Stiffness on Glioblastoma Cell Behavior Using PEG-based Hydrogels. *Molecular Pharmaceutics* **11** (7), 2115–2125.

Watanabe, S. 2010 Asymptotic Equivalence of Bayes Cross Validation and Widely Applicable Information Criterion in Singular Learning Theory. *Journal of Machine Learning Research* **11** (116), 3571–3594.

Weisstein, E. W. 2004 Chi Distribution. https://mathworld.wolfram.com/ChiDistribution.html, accessed: 28/05/2021.

Wheeler, J. 2020 Navigation of surface-motile bacteria in developing bacterial biofilms. PhD thesis, University of Oxford.

Wolf, K., Lindert, M., Krause, M., Alexander, S., Riet, J., Willis, A. L., Hoffman, R. M., Figdor, C. G., Weiss, S. J. & Friedl, P. 2013 Physical limits of cell migration: Control by ECM space and nuclear deformation and tuning by proteolysis and traction force. *Journal of Cell Biology* **201**, 1069–1084.

Wu, P., Gilkes, D. M. & Wirtz, D. 2018 The Biophysics of 3D Cell Migration. *Annual Reviews of Biophysics* **47**, 549–567.

Wu, P., Giri, A., Sun, S. X. & Wirtz, D. 2014 Three-dimensional cell migration does not follow a random walk. *PNAS* **111** (11), 3949–3954.

Wu, P., Giri, A. & Wirtz, D. 2015 Statistical analysis of cell migration in 3D using the anisotropic persistent random walk model. *Nature Protocols* **10** (3), 517–527.

Xie, L., Altindal, T., Chattopadhyay, S. & Wu, X. 2010 Bacterial flagellum as a propeller and as a rudder for efficient chemotaxis. *PNAS* **108** (6), 2246–2251.

Yamada, K. M. & Cukierman, E. 2007 Modeling Tissue Morphogenesis and Cancer in 3D. *Cell* **130**, 601–610.

Yang, Y., He, J., Altindal. T., Xie & L., Wu, X. 2015 A Non-Poissonian Flagellar Motor Switch Increases Bacterial Chemotactic Potential. *Biophysical Journal* **109** (5), 1058–1069.

Ye, K., Yang, X., Ji, Y. & Wang, M. 2020 A system for determining maximum tolerated dose in clinical trial. *Statistical Theory and Related Fields* .

Youngflesh, C. 2018 MCMCvis: Tools to visualize, manipulate, and summarize MCMC output. *Journal of Open Source Software* **3** (24), 640.

Yurchenko, I., Vensi Basso, J. M., Syrotenko, V. S. & Staii, C. 2019 Anomalous diffusion for neuronal growth on surfaces with controlled geometries. *PLoS ONE* **14** (5), e0216181.

Zaman, M. H., Kamm, R. D., Matsudaira, P. & Lauffenburger, D. A. 2005 Computational Model for Cell Migration in Three-Dimensional Matrices. *Biophysical Journal* **89**, 1389–1397.

Zaman, M. H., Matsudaira, P. & Lauffenburger, D. A. 2007 Understanding Effects of Matrix Protease and Matrix Organization on Directional Persistence and Translational Speed in Three-Dimensional Cell Migration. *Annals of Biomedical Engineering* **35** (1), 91–100.

Zaman, M. H., Trapani, L. M., Sieminski, A., Mackellar, D., Gong, H., Kamm, R. D., Wells, A., Lauffenburger, D. A. & Matsudaira, P. 2006 Migration of tumor cells in 3D matrices is governed by matrix stiffness along with cell-matrix adhesion and proteolysis. *PNAS* **103** (29), 10889–10894.

Zwiers, F. W. & von Storch, H. 1995 Taking Serial Correlation into Account in Tests of the Mean. *Journal of Climate* **8** (2), 336–351.

Appendix A

MATLAB Code for running the PRW framework in 2 and 3 dimensions

All code for running the framework on both the *in silico* and experimental data sets can be found for the 3D work at

https://github.com/m-scott22/PRW3DCellMotilityFramework

and for the 2D work at

https://github.com/m-scott22/PRW2DCellMotilityFramework

A detailed workflow for how the code should run, what the variables mean and what outputs are returned is shown in figure A.1.

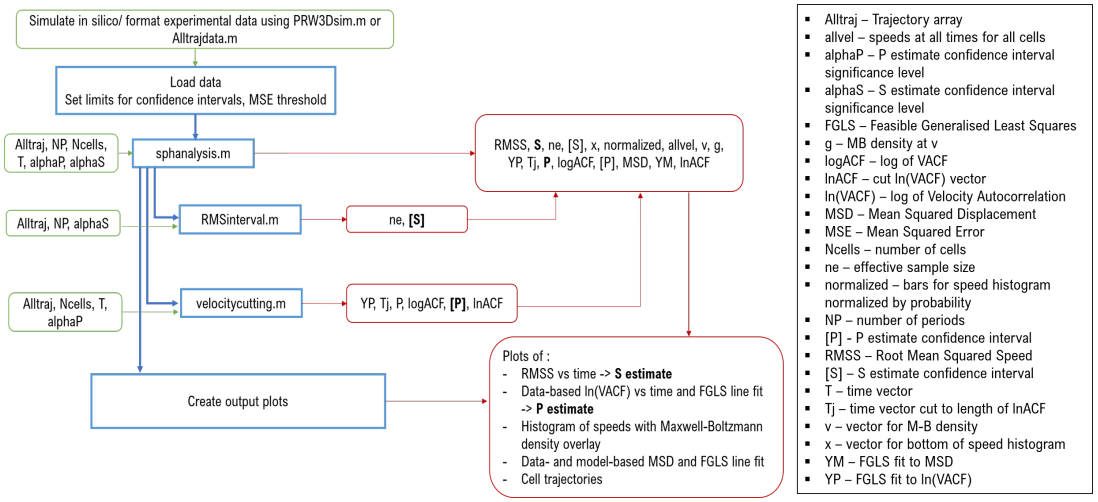

Figure A.1: **Detailed schematic of the workflow for the framework**

Appendix B

JAGS model code for Bayesian parameter estimation

B.1 Example JAGS code for 'Estimating S alone'; AR(1) model

```
model {
#Priors
mu ~ dnorm(0, 0.01);
tau.pro ~ dgamma(0.001,0.001);
sd.pro <- 1/sqrt(tau.pro);
phi ~ dunif(-1, 1);
S <- mu/(1.0-phi);

#Model
predY[1] <- Y[1];
for(i in 2:N) {
predY[i] <- mu + phi * Y[i-1];
Y[i] ~ dnorm(predY[i], tau.pro);}}
```

B.2 Example JAGS code for 'Estimating P alone'; AR(1) model

```
model {
#Priors
tau.obs ~ dgamma(0.001,0.001);
sd.obs <- 1/sqrt(tau.obs);
phi ~ dunif(-1,1);
tau.cor <- tau.obs / (1-phi*phi);
a ~ dunif(-1, 1);
b <- -1.0/P;
P ~ dgamma(4,0.25);

#Model
mu[1] <- a + b * x[1];
epsilon[1] <- Y[1] - mu[1];
predY[1] <- mu[1]; # initial value
for(i in 2:N) {
mu[i] <- a + b * x[i];
predY[i] <- mu[i] + phi * epsilon[i-1];
Y[i] ~ dnorm(predY[i], tau.cor);
epsilon[i] <- (Y[i] - mu[i]) - phi*epsilon[i-1];}}
```

B.3 Example JAGS code for 'Estimating P and S simultaneously - S prior informed'; AR(1) model

```
model {
# Priors
```

```
tau.obs ~ dgamma(0.001,0.001);
sd.obs <- 1/sqrt(tau.obs);
phi ~ dbeta(2,2);
tau.cor <- tau.obs / (1-phi*phi);
a <- 2*log(S)
b <- -1.0/P
S ~ dgamma(77,78) #prior from RMSS
P ~ dgamma(4,0.25)

#Model
mu[1] <- a + b * x[1];
epsilon[1] <- Y[1] - mu[1];
predY[1] <- mu[1]; # initial value
for(i in 2:N) {
mu[i] <- a + b * x[i];
predY[i] <- mu[i] + phi * epsilon[i-1];
Y[i] ~ dnorm(predY[i], tau.cor);
epsilon[i] <- (Y[i] - mu[i]) - phi*epsilon[i-1];}}
```

B.4 Example JAGS code for 'Estimating P and S simultaneously - S prior informed'; AR(2) model

```
model {
#Priors
tau.obs ~ dgamma(0.001,0.001);
sd.obs <- 1/sqrt(tau.obs);
phi1 ~ dbeta(2,2);
phi2 ~ dunif(-1,1);
tau.cor <- tau.obs/(1-phi1*phi1-phi2*phi2);
```

```
S ~ dgamma(77,78); #Informed Prior
P ~ dgamma(4,0.25);
a <- 2*log(S);
b <- -1.0/P;

#Model
mu[1] <- a + b * x[1];
mu[2] <- a + b * x[2];
epsilon[1] <- Y[1] - mu[1];
epsilon[2] <- Y[2] - mu[2];
predY[1] <- mu[1];
predY[2] <- mu[2];
for(i in 3:N) {
mu[i] <- a + b * x[i];
predY[i] <- mu[i] + (phi1 * epsilon[i-1]) +
(phi2 * epsilon[i-2]);
Y[i] ~ dnorm(predY[i], tau.cor);
epsilon[i] <- (Y[i] - mu[i]) -
(phi1*epsilon[i-1])- (phi2*epsilon[i-2]);}}
```

Appendix C

R code for JAGS MCMC simulations and model selection

C.1 Example JAGS code for 'Estimating S alone'; AR(1) model

```
library('rjags')
library(coda)
#Read in data to fit model to
data<-read.csv(file='insilicodata11001.csv',
header=TRUE, sep=",")
RMSS<-data$RMSS

#Define initial values
model.inits <- list(S=1, phi=0.5, tau.pro=1)

#Choose burn-in and number of iterations
iterations <- 50000
burnin <- floor(3*iterations/4)
```

```
#Specify JAGS model
model.fit <- jags.model(file=
"correlatedobs11001test.txt",
data=list('N'=1001, 'Y'=RMSS),
inits=model.inits, n.chains = 2,
n.adapt = 100)

#Get MCMC samples and choose which nodes to
monitor
model.samples <- coda.samples(model.fit,
c("phi","mu","sd.pro","S"), n.iter=iterations)

#Obtain summary estimates of the parameters
summary(window(model.samples, start = burnin))

#Plot posterior distributions
plot(model.samples, trace=TRUE, density = TRUE)
```

C.2 Example JAGS code for 'Estimating P and S simultaneously - S prior informed'; AR(1) and AR(2) model with model selection

```
library('rjags')
library(coda)
#Read in data to fit model to
data<-read.csv(file='insilicodata11001.csv',
header=TRUE, sep=",")
names(data)[1]<-"logACF"
```

```
lnACF <- data$lnACF
t<-data$Tj
n<-59
model.inits <- list(S=1, P=1, phi=0.5,
tau.obs=1)

#AR1
#Choose burn-in of half the iterations
iterations <- 200000
burnin <- floor(iterations/2)
thin<-10

model.try<-autorun.jags(
"SPcorrelatederrors11001s.txt",
monitor=c("S","P","sd.obs","phi","dic"),
 data=list('N'=n,
'Y'=lnACF,'x'=t),n.chains=4,
inits=model.inits,
startburnin=burnin,thin=thin)

#Specify JAGS model
model.fit <- jags.model(file=
"SPcorrelatederrors11001s.txt",
data=list('N'=n, 'Y'=lnACF,'x'=t),
inits=model.inits,
n.chains = 4, n.adapt = 100)

#Obtain summary estimates of the parameters
summary(model.try)

#Plot posterior distributions
```

```
plot(model.try,c("trace","histogram"))

#Get WAIC
load.module("dic")
s <- jags.samples(model.fit, c("WAIC"),
type="mean",
n.iter=100000, thin=thin, n.chains=4)
waicoutput1<-sapply(s,sum)
sest <- jags.samples(model.fit, c("S", "P",
"sd.obs"),
type="trace", n.iter=100000, thin=thin,
n.chains=4)


#Get DIC
extract(model.try,"dic")

###############


#WAIC

Pvals1<-as.numeric(sest[["P"]])
Svals1<-as.numeric(sest[["S"]])
sdvals1<-as.numeric(sest[["sd.obs"]])

meanoversamples1 <- NULL
for(j in 1:n){
ldensity1<- (1/(sdvals1*sqrt(2*pi)))*
exp(-(lnACF[j]-2*log(Svals1)+(t[j]/Pvals1))
^2/(2*sdvals1^2))
```

```
meanoversamples1[j]<-mean(ldensity1)
}
logmeandensity1<-log(meanoversamples1)

lppd1<-sum(logmeandensity1)
penalty1<-waicoutput1[1]
WAIC1<--2*(lppd1-penalty1)


######################
#AR2

#Specify JAGS model

model.inits2 <- list(S=1, P=1, phi1=0.5,
phi2=0.5, tau.obs=1)


model.try2<-autorun.jags(
"SPcorrelatederrors211001s.txt",
monitor=c("S","P","sd.obs","phi1","phi2","dic"),
data=list('N'=n, 'Y'=lnACF,'x'=t),
n.chains=4,inits=model.inits2,startburnin=burnin,
thin=thin)

model.fit2 <- jags.model(file=
"SPcorrelatederrors211001s.txt",
data=list('N'=n, 'Y'=lnACF,'x'=t),inits=
model.inits2, n.chains = 4, n.adapt = 100)

#Obtain summary estimates of the parameters
```

```
summary(model.try2)

#Plot posterior distributions
plot(model.try2,c("trace","histogram"))


#Get DIC
extract(model.try2,"dic")

#Get WAIC
load.module("dic")
s2 <- jags.samples(model.fit2, c("WAIC"),
type="mean", n.iter=414140, thin=thin,
 n.chains=4)
waicoutput2<-sapply(s2,sum)
sest2 <- jags.samples(model.fit2, c("S", "P",
"sd.obs",
"phi1","phi2"), type="trace", n.iter=414140,
thin=thin, n.chains=4)

Pvals2<-as.numeric(sest2[["P"]])
Svals2<-as.numeric(sest2[["S"]])
sdvals2<-as.numeric(sest2[["sd.obs"]])

meanoversamples2 <- NULL
for(j in 1:n){
ldensity2<- (1/(sdvals2*sqrt(2*pi)))*
exp(-(lnACF[j]-2*log(Svals2)+(t[j]/Pvals2))^2
/(2*sdvals2^2))
meanoversamples2[j]<-mean(ldensity2)
}
```

```
logmeandensity2<-log(meanoversamples2)

lppd2<-sum(logmeandensity2)
penalty2<-waicoutput2[1]
WAIC2<--2*(lppd2-penalty2)

#Bayes Factor in favour of AR1
library(logspline)
phi2vals<-as.numeric(sest2[["phi2"]])
posterior_phi2<-logspline(phi2vals,-1,1)
posterior_0 <- dlogspline(0, posterior_phi2)
prior_0<-0.5
BAYESFACTOR<-posterior_0/prior_0

save(model.try,model.try2,s,sest,s2,sest2,WAIC1,
WAIC2,BAYESFACTOR,file="finalsamples.Rdata")
```

Appendix D

Detailed explanation of the Chi-square test in the context of studying twiddles

For clarity, an example of the chi-square test of independence is studied here in more detail. To demonstrate this we will use the example data used to test hypothesis 2 in the Pooled control case for manual paired data, shown in table D.1.

OBSERVED	Not correct exit	Correct exit	Total
Not correct entry	86	64	150
Correct entry	43	31	74
Total	129	95	224

Table D.1: Table of observed values

The chi-square test looks to compare observed values with what we would expect to see if there was independence. The expected values are estimated by multiplying row and column totals and dividing by the overall total. Doing so for this data gives expected values as shown in table D.2.

To conduct the test one must calculate Pearson's chi-square statistic

$$X^2 = \sum_i \frac{(O_i - E_i)^2}{E_i},$$

EXPECTED	Not correct exit	Correct exit	Total
Not correct entry	86.4	63.6	150
Correct entry	42.6	31.4	74
Total	129	95	224

Table D.2: Table of expected values

where O_i is the observed data from cell i, E_i is the expected data from cell i and the sum is over all cells in the table. This value is then compared to the critical value from the chi-square distribution with $(r-1) \times (c-1)$ degrees of freedom when there are r rows and c columns in the table. If the value of the test statistic is bigger than the critical value the null hypothesis is rejected and we say that there is some association or dependence between these categorical variables.

In the context of the surface-attached bacteria work we are looking for independence between entry to, and exit from twiddles. So we are looking for p-values to be greater than 0.05, and values of the chi-square statistic to be less than the critical values from the χ^2_1 distribution, as all tables have 2 rows and 2 columns.

But what does this mean intuitively? We want to ensure that one variable doesn't influence the other, and that the relative frequencies of one variable are the same over all the levels of the other (van den Berg, R. G., 2021). Practically this means that if, for example, the overall probability of a correct exit is 50%, then this probability should remain at about 50% regardless of whether the entry into the twiddle was correct or not correct. This would be the same if we considered the overall probability of a correct or not correct entry along with either manner of exit as the chi-square test is looking for associations between the variables and does not take into account that entry comes before exit in this context.

So, going back to the observed data in table D.1, the overall probability of a correct exit is $95/224 \approx 42.4\%$. Now if we assume that entry into the twiddle is correct, the probability of a correct exit given this correct entry is $31/74 \approx 41.9\%$. If we assume entry is not correct, the probability of a correct exit given not correct entry is $64/150 \approx 42.7\%$. We can see that regardless of the manner

of entry, the probability of a correct exit remains approximately constant. If this remains the case for all combinations of exit and entry, then we can be sure that neither variable is affecting the other and we have independence. If an association or dependence is observed, this means that the values of one of the categorical variables differs depending on the values of the other.

It is important to note, particularly in this context where there is a bias for correct exits from twiddles in a gradient, that we can still see more correct exits than not whilst maintaining independence from manner of entry. This means there could be something as extreme as a 90% probability of a correct exit, but still every chance that exit and entry are independent from one another, as long as the manner of entry doesn't affect this probability significantly. In other words, the chi-square test only demonstrates independence between the two variables and does not mean that the probabilities of each category within the variable should be equal.

We note, finally, that the observed values in table D.1 and the expected values in table D.2 are remarkably similar. It is thus clear to see why we do not obtain a significant result upon conducting the chi-square test on this data, and we conclude that entry and exit are independent.

Appendix E

Information relevant to hypothesis tests on tumbles in surface-attached bacteria

Hypothesis 2 Sample Sizes - *Manual Data*					
	00	**01**	**10**	**11**	**Total**
Control					
Experiment 1	23	17	21	6	67
Experiment 2	31	23	10	12	76
Experiment 3	32	24	12	13	81
Pooled	86	64	43	31	224
Gradient					
allDMSOWT	24	19	10	12	65
Experiment 2	18	30	13	12	73
Experiment 3	21	24	18	18	81
Pooled	63	73	41	42	219

Table E.1: Samples sizes for tests of hypothesis 2 on the manual data. 00 = not correct entry and not correct exit, 01 = not correct entry and correct exit, 10 = correct entry and not correct exit, 11 = correct entry and correct exit.

Hypothesis 3 Sample Sizes - *Manual Data*					
	00	**01**	**10**	**11**	**Total**
Control					
Experiment 1 correct entry	18	3	5	1	27
Experiment 2 correct entry	7	3	10	2	22
Experiment 3 correct entry	7	0	1	1	9
Pooled correct entry	32	6	16	4	58
Experiment 1 not correct entry	10	1	7	1	19
Experiment 2 not correct entry	9	6	8	1	24
Experiment 3 not correct entry	4	3	7	0	14
Pooled not correct entry	23	10	22	2	57
Gradient					
Experiment 1 correct entry	7	2	3	9	21
Experiment 2 correct entry	11	2	5	7	25
Experiment 3 correct entry	15	3	4	14	36
Pooled correct entry	33	7	12	30	82
Experiment 1 not correct entry	7	4	7	3	21
Experiment 2 not correct entry	6	3	18	1	28
Experiment 3 not correct entry	7	3	7	4	21
Pooled not correct entry	20	10	32	8	70

Table E.2: Samples sizes for hypothesis 3 tests on the manual data. 00 = not correct exit and no reversal, 01 = not correct exit and reversal, 10 = correct exit and no reversal, 11 = correct exit and reversal.

Appendix F

Obtaining the equilibrium orientation distribution for reversals

Consider constructing a solution to the system defined by equations 4.5 and 4.6, f_R, such that

$$\lambda f_R = \int_0^{2\pi} \lambda(\theta')K(\theta,\theta')f_R(\theta')\,d\theta', \tag{F.1}$$
$$K(\theta,\theta') = h(|\theta-\theta'|),$$

for some smooth function h acting as turn kernel K. Since f_R will be periodic, we can write down its Fourier series expansion as

$$f_R = f(\theta)\sum_k C_k\cos(k\theta) + D_k\sin(k\theta),$$

where $f(\theta)$ is the stationary orientation distribution given by equation 4.7.

Putting this into equation F.1 gives

$$\begin{aligned}\sum_k C_k\cos(k\theta) + D_k\sin(k\theta) &= \int_0^{2\pi} h(|\theta-\theta'|)\left(\sum_k C_k\cos(k\theta') + D_k\sin(k\theta')\right)d\theta' \\ &= \sum_k \alpha_k\left(\sum_k C_k\cos(k\theta) + D_k\sin(k\theta)\right), \end{aligned}\tag{F.2}$$

where $\alpha_k = 2\int_0^\pi \cos(ku)h(u)\,du$ are the moments of the turn kernel K.

It is clear that for this equation to be satisfied, either $\alpha_k = 1$, for all k, or $C_k = D_k = 0$ for all $k > 0$, since when $k = 0, \alpha_0 = 1$ and the equation is satisfied. The only way that $\alpha_k = 1$ is if there is no change in a cell's direction, and this is clearly not applicable to a situation where reversals or twiddles occur. Thus we must take $C_k = D_k = 0$ for $k > 0$, and thus we are left with

$$f_R = f(\theta)C_0 \implies f(\theta) = \frac{f_R}{C_0}.$$

Given that $\int_0^{2\pi} f(\theta)d\theta = 1$, we deduce that $C_0 = 1$ and thus the final result is that $f(\theta) = f_R$, i.e. that the equilibrium orientation distribution $f(\theta)$ is the solution to the system.

Appendix G

Code for simulating twiddles and reversals of surface-attached bacteria as per the proposed IBM

Code to simulate both the Reversal only and Twiddle only models can be found at the following link:
https://github.com/m-scott22/Surface-attached-IBM